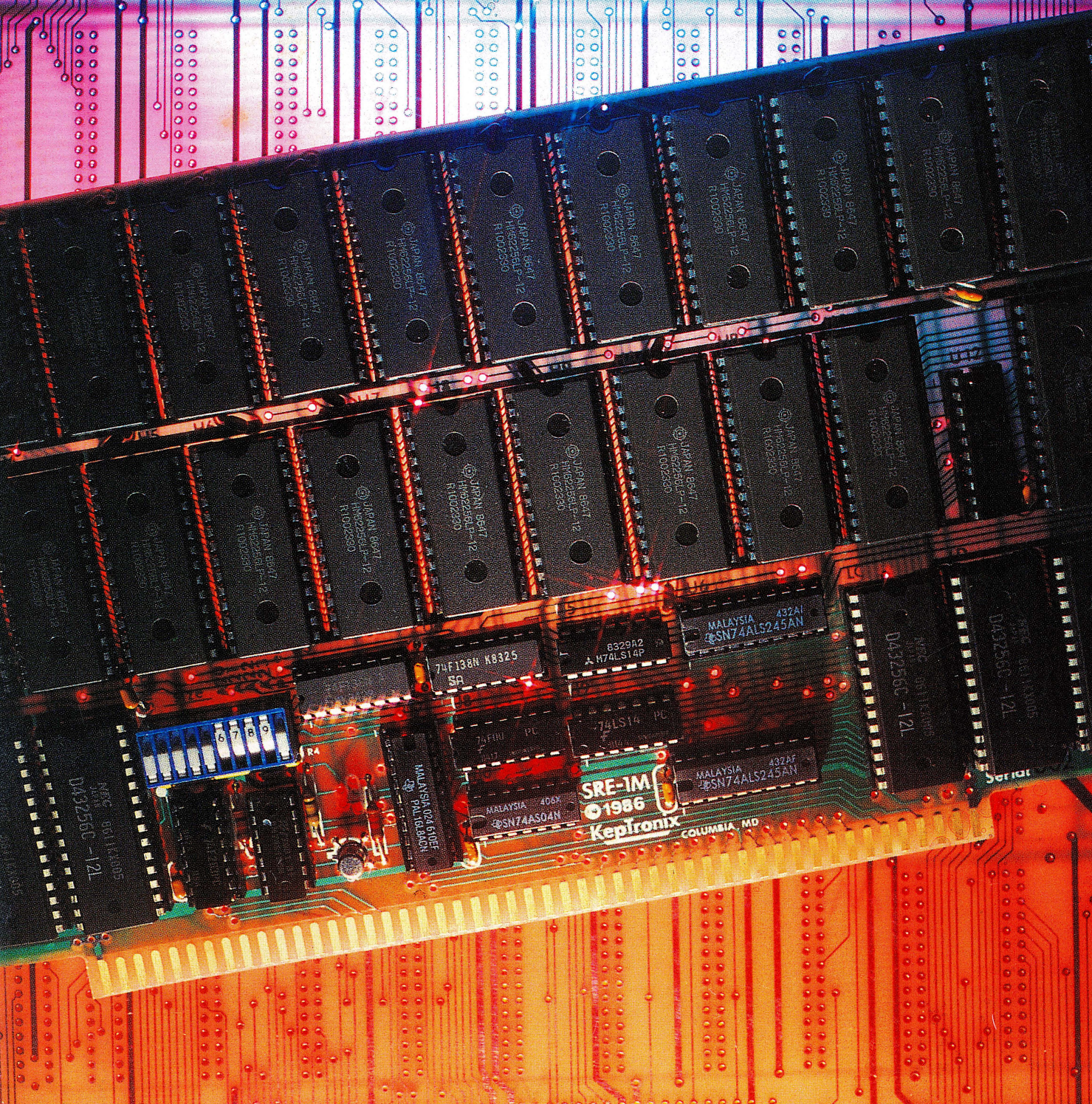
$4.90 VOL. 1, NO. 5 • WINTER 1987

S-100 journal



Bus Specific Networks

DOS-IT-YOURSELF

At last, you can build PC-DOS compatible S-100 bus systems — only with ICM's new CPS-MS Slave Cards.

Only InterContinental Microsystems puts PC-DOS and MS-DOS capabilities on S-100. New compatibility with no LAN hookups. Now simply add the new CPS-MS Slave Card and a PC terminal to your S-100 bus system, and you've added a PC. For every CPS-MS card and terminal you add, you get an additional PC — with full connectivity under TurboDOS. Our MS/1000 CPS-MS package, including ICM's exclusive MS/1000 software, saves space, installs easier, reduces noise, reduces heat, reduces maintenance and repair costs, is easier to integrate and performs twice as fast as an IBM PC or XT.

You'll love the "low-cost expansion/add-on performance ratio" of ICM's new CPS-MS cards. They come as a total IBM-compatible package complete with 16-bit processor, M/STER PC-emulating terminal with ICM's monochrome graphics, PC-DOS, TurboDOS/PC, and ICM's exclusive MS/1000 software.

Call ICM for more information and reseller pricing.

IBM PC-AT, IBM PC-XT, IBM PC and PC-DOS, are registered trademarks of International Business Machines Corp.
MS-DOS is a registered trademark of Microsoft Corp. TurboDOS/PC is a registered trademark of Software 2000 Corp.

PC, ARCNET, AND S-100 APPLICATIONS

ICM

TOMORROW'S BREAKTHROUGHS TODAY

InterContinental Microsystems
4015 Leaverton Court, Anaheim, CA 92807
Phone: (714) 630-0964
Telex: 821375 SUPPORT UD
Easylink: 62562040 BBS (714) 632-8750

S-100 journal

VOL. 1 NO. 5 WINTER 1987

Editor and Designer
Jay Vilhena

Associate Editor
Robert Petersen

Technical Editor
Don Pannell

Circulation
Linda Vilhena

Advertising
Jay Vilhena

Software Consultant
Steve Renshaw

Contributing Artists
A. & G. N. Pereira
Joslyn L. Johanna

ISSN 0892-6190

S-100 Journal is published quarterly by Octoplus Corporation, 1275 N. University Ave., Unit 7, Provo, UT 84604. Telephone (801) 373-0696.

Copyright©1986 by Octoplus Corporation. All rights reserved.

Subscriptions: U.S.A.: $14.00 for 1 year, $25.00 for 2 years. Canada and Mexico: add $3.00 per year. Other countries: add $6.00 per year. All foreign payments must be in U.S. currency. Send orders and payments to S-100 Journal, P.O. Box 1914, Orem, UT 84057.
Advertising: Contact Jay Vilhena at one of above addresses or call (801) 373-0696.
Postmaster: Send address changes to S-100 Journal, P.O. Box 1914, Orem, UT 84057.

outline

22 BARCODE MAGIC FOR THE S-100 SYSTEM

29 HARDWARE PRIMER ON S-100 BASED LANS

38 DESIGN OF 1-MEGABYTE STATIC MEMORY BOARD

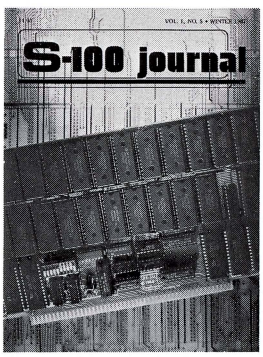
63 32-BIT EXPANSION TO THE S-100 BUS

67 THE MACROTECH ADIT INTELLIGENT I/O BOARD

4 EDITORIAL
7 EDITOR INTERFACE
8 696 BUS
15 MULTIUSER OS
18 HOMEBREWING
54 BITS
72 NEWS AND NEW PRODUCTS
75 S-100 SOFTWARE
76 S-100 DIRECTORY

OUR COVER

S-100 static RAM boards with one or more Megabytes of memory are becoming popular. In this issue, we show you how to design your own Megabyte board. See page 38.



What you should know about the S-100 market & Advanced Digital... BEFORE you buy;

"With the new wave of outstanding S-100 products hitting the market, S-100 manufacturers continue to assert their dedication to superior quality and innovative design. For many years ahead, they will equip thousands of small businesses and professionals with the most versatile and upgradable microsystems in existence."

—Jay Vilhena, Editor
S-100 Journal

"At Advanced Digital Corporation our business philosophy can be stated in one word - "Pride." Pride in the quality and performance of the products we design and manufacture...Pride in the dedication, experience and performance of our employees...and Pride in our achievements and contributions to the S-100 marketplace."

—Hossein Asadi, President
Advanced Digital Corporation

1. Attractive Features. Attractive Prices.

Advanced Digital Corporation offers the most complete line of S-100 products with the latest in the state-of-the-art microcomputer technology. All of ADC's S-100 products are powerful high speed 8-bit and 16-bit single board computers. Your S-100 system can grow from ordinary to awesome with ADC's superb family of S-100 boards quickly, easily and cost-effectively.

2. Financial Stability

Founded in 1980, Advanced Digital Corporation is one of the most financially sound, privately held computer manufacturers. The company has consistently shown a profit in every quarter since 1980. It is the intention of ADC to continue as the industry leader and remain in the forefront of technology assuring you of the finest product and service for years to come.

3. Quality Assurance

Rigorous testing of each and every product by ADC's highly skilled technicians assures customers of trouble-free products. ADC prides itself with the strictest quality control in the industry. No company has higher quality control standards, or a better success rate than ADC's. Every product is backed by a one year warranty.

4. Customer Service; 24 Hour Repair.

Highly-skilled technicians are assigned to customer support full time. In most cases, a problem can be solved over the phone in a few minutes. If not, repairs can be completed within 24 hours from receipt of product. Centralized repair service is available through ADC's Service Center.

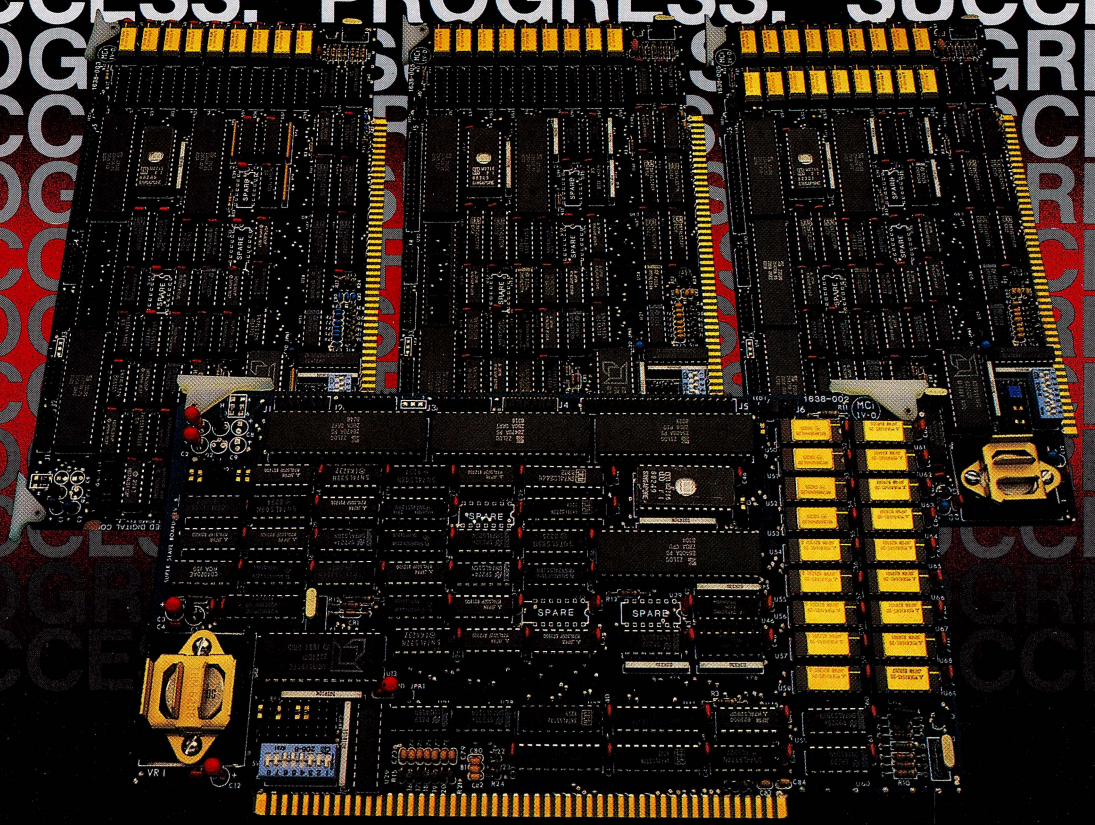
5. Market Penetration; Over 12,000 units installed.

With an installed base of over 12,000 units worldwide, ADC provides service to thousands of businesses and individual owners. By developing a superior line of S-100 products, ADC has become a major force in the industry. Customers include: US Navy, NASA, Harris Systems, Aerospace Corp.

6. Resources... Technical Expertise, Advanced Engineering.

ADC has assembled a team of highly experienced, industry renown specialists in engineering, manufacturing, marketing, sales and administration. ADC is a company of pioneers and innovators that have led the way and set the S-100 standards.

PROGRESS. SUCCESS. PROGRESS.
SUCCESS. PROGRESS. SUCCESS.



Founded in 1980, by a team of highly skilled and experienced engineers, Advanced Digital Corporation has grown to become a leading supplier of Single-Board Computers, Multi-User, Multi-Processor computer systems and memory subsystems for the S-100 Bus and PC Compatible Products marketplace. We're a company that has made tremendous growth without sacrificing the major factor that led to it. Quality...quality in our product, in our people, and in the way we do business.

Leading our success in the S-100 market, is a legacy of products that represent the needs of the S-100 market. Products such as Super System II, Super 16, Super 186 and Multi-Slave have made ADC the success it is today.

SUCCESS STORIES:

- SUPER SYSTEM II...A fully integrated system that runs under CP/M 2.2® for single-user applications or TurboDOS® for multi-user installations for both 8 and or 16-Bit including networking to PC's.
- SUPER 16...The cost effective 16-Bit 8 MHz Multi-User Single Board Slave processor.
- SUPER 186... The first high speed 8 MHz 16-Bit S-100 Single Board Computer.
- MULTI-SLAVE...A three user board contains three independent computers at 8 MHz and runs the TurboDOS operating system.

Progress. Success. It's the Story of our Company.

The success of ADC in the S-100 market is also due to the following line of quality products: • Super Star • Super Quad • Super Six • Super Slave • Master 9.

At ADC we're an aggressive company, a progressive company, an innovative company. To be successful, we know that each product we develop must be totally unique to the market.

Our progress and success is with products that have gained the highest ratings and the respect of the entire S-100 industry. Our success is based on superior technology, unique features, and the most cost competitive prices in the market. When you're looking for progress, look to ADC, because it's the story of our company.

ADC
ADVANCED DIGITAL CORPORATION

ADVANCED DIGITAL CORPORATION
5432 Production Drive • Huntington Beach, CA 92649
(714) 891-4004 • (800) 251-1801 • Telex 183210 ADVANCED HTBH

ADVANCED DIGITAL U.K. LTD.
27 Princes Street, Hanover Square • London W1R8NQ-United Kingdom
(01) 409-0077 • (01) 409-3351 • TLX 265840 FINEST

© TurbosDOS is a trademark of Software 2000. ® CP/M 2.2 is a trademark of Digital Research Corporation.

SEE US AT FALL/COMDEX 86 • BOOTH 1480

editorial

HARK! KILL THE MONSTERS AND SAVE THE MAIDEN

Despite its many years of success, the S-100 bus still battles two gruesome internal enemies: the existence of incompatible boards and the scantiness of drivers. Fortunately, the same modularity that breeds the monsters also nurtures the most precious pearl: the **ABILITY** to add boards, swap components, run several CPUs, upgrade to faster speeds, replace microprocessors, go multiuser, build LANs, change terminals, add terminals, have graphics, get huge or small amounts of RAM, shape the number and type of I/O ports, buy RAMdisks, build clockboards, update operating systems, discard old operating systems but keep the applications software, run several operating systems simultaneously, store data in hard disks, backup with tape, look at the schematics of the CPU board, have 8 bits, 16 bits, and/or 32 bits, choose among Intel, Motorola, Zilog, NEC, Hitachi, or whatever CPU line we prefer. We have the **ABILITY**, the freedom of being in direct control of our systems, of having an open box, of making our computers feature the exact parts and performance that **WE** want or need.

Recently, in a regional publication, I wrote an article singing the praises of S-100 computers. I had several calls from readers who had never heard of our wonder-machines. One reader asked: 'Does it have multiuser and multitasking?' A maze of pictures flashed through my head: a dozen S-100 computers with only multiuser, another dozen with only multitasking, another dozen with both, still another dozen with neither. I hesitantly answered, 'Yeah! Sure.' He replied: 'What do you mean 'Yeah! Sure.'? Does it or does it not?'

The microcomputer consumer is used to entering a computer store, pointing at a machine, or a picture of one, and asking for the features. At which time a well-groomed salesperson with a remarkable lack of computer knowledge proceeds to unreel a set list of features stored in a ROM-like brain. The concept of a computer that grants the purchaser the **ABILITY** to 'dial' the features that he or she wants is not of garden variety. It is S-100's most cherished quality; it is also fragile.

ABILITY is the name of the maiden that we must protect. At one time or another, for reasons best known to themselves, some S-100 companies have locked their S-100 boards inside a box, erased all traces of the word 'S-100' from their literature and marketed their systems like the busless dead-end boxes that proliferate throughout the industry. This is a brutal mistake! It not only hurts those companies in the long run, it is detrimental to the S-100 community as a whole by making our installed base appear smaller than it actually is.

Even OEMs and other integrators who sell complete systems to vertical markets should always state with pride that their systems are based on the S-100 bus, the most modular, dependable, and upgradable computer in existence. It is time to shed the winding sheet of shyness that has enveloped the S-100 community for the last 3 or 4 years. The S-100 bus is superlative and peerless; let the world know it.

Nevertheless, we must still battle the monsters. S-100 board designers and manufacturers must pay careful consideration to ensuring that their products are compatible not only with their own previously-released boards, but also with a maximum of products from other manufacturers. There is still too much incompatibility in the field. I am referring to contemporary boards and not to the natural process of new boards not working with much-earlier-generation versions. It obviously cannot be expected that an old 2-MHz card work reliably or at all with 12-MHz products, but contemporary boards must. S-100 buyers can help here too. When planning to purchase a board, call the company first and ask about compatibility with your existing boards. You should not always expect a precise answer since it may not be easy for the person on the other end to ascertain your hardware environment, but you should detect a willingness to help. If not, buy from another company.

Finally, there is a flagrant lack of software drivers. Nearly all S-100 boards on the market would probably sell three or four times as many units if drivers were available. For various reasons, hardware vendors tend to stick to one or two favorite operating systems and only supply drivers for those. To help solve the problem, vendors should at least try to identify and purchase drivers that customers might have written to run their boards under other operating systems and/or hardware, and make these drivers available when selling the cards. S-100 Journal will help too. We will pay well for articles that show how to run popular boards under different environments.

Jay Vilhena

The Peak of Performance

SCALE THE HEIGHTS OF PRODUCTIVITY

Sure, you've proven that in your hands a computer is a productive tool. But if you haven't teamed up with a SemiDisk you have heights yet to climb!

IT'S NO MERE RAMDISK

SemiDisk has been leading the way for Disk Emulators since their inception. If you've seen RAMdisks you know what it's like to load programs in an instant, and read or write files without delay. Unlike alternatives, the SemiDisk offers up to 8 megabytes of instant-access storage while leaving your computer's main memory free for what it does best - computing!

KEEP A GRIP ON DATA

Go ahead, turn off your computer. Take a vacation. With the battery backup option, your valuable data will be there in the morning even if you aren't. You'll sleep better knowing not even a 5 hour blackout will sabotage your files.

NEW LOWER SEMIDISK PRICES THAT WON'T SNOW YOU UNDER

	512K	2Mbyte
IBM PC, XT, AT	$495	$995
Epson QX-10	$595	$995
S-100, SemiDisk II	$799	$1295
S-100, SemiDisk I	$595	———
TRS-80 II, 12, 16	$695	$1295
Battery Backup Unit	$130	$130

Software drivers available for CP/M 80, MS-DOS, ZDOS, TurboDOS, and VALDOCS 2.

SEMIDISK

SemiDisk Systems, Inc.
P.O. Box GG, Beaverton, Oregon 97075
503-626-3104

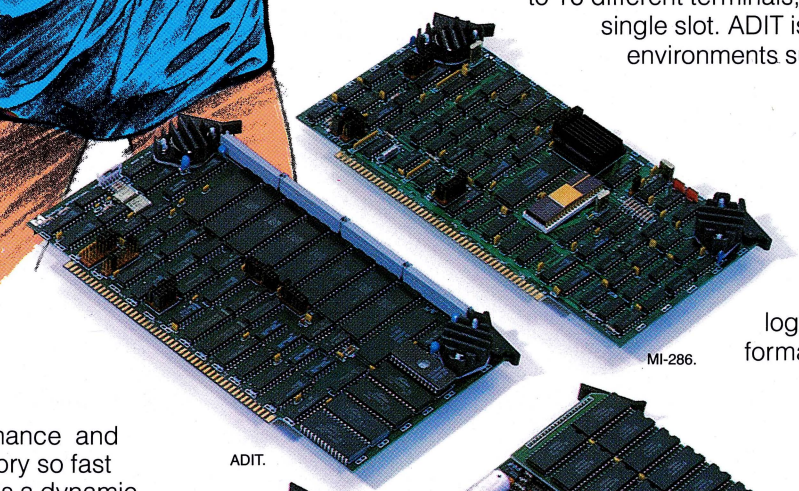
Call 503-646-5510 for CBBS/NW, and 503-649-8327 for CBBS/Aloha, all SemiDisk equipped computer bulletin boards, (300/1200/2400 baud) SemiDisk, SemiSpool trademarks of SemiDisk Systems.

MACROTECH—STILL THE S-100 PERFORMANCE PACESETTER

MI-286. Our 80286/Z80H Dual CPU Board is at least twice as fast as Compupro's 8085/88 and it's a direct replacement. The MI-286 has already become the standard by which other 80286 based systems are measured. Ask us for a complimentary Benchmark Report.

ADIT. There's nothing else like it on the market. It's an Intelligent I/O Board with its own real time firmware that lets you control up to 16 different terminals, modems or printers all from a single slot. ADIT is the performance standard in environments such as Alpha Micro where I/O speed is critical.

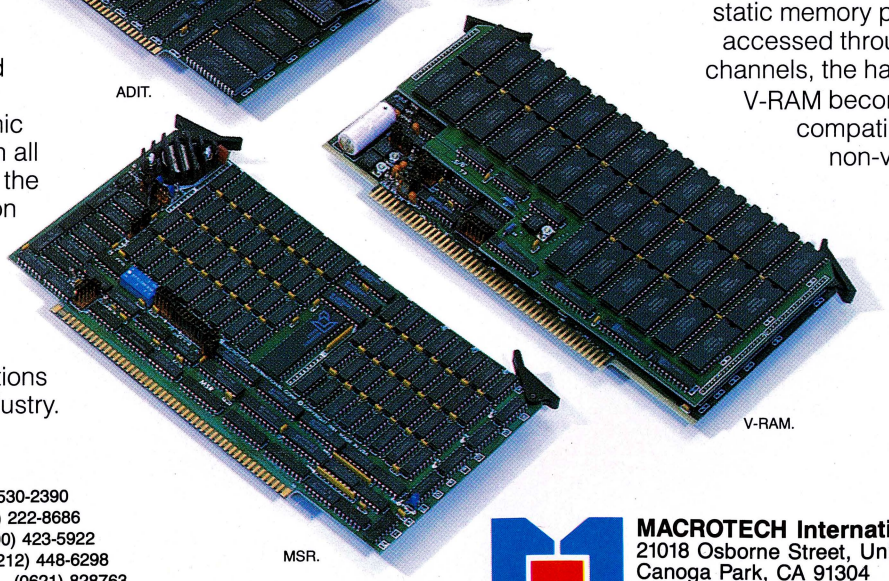
V-RAM. High performance Static CMOS system memory/virtual disk in either quarter or half megabyte configurations. With its on-board battery and power-fail logic, the V- RAM sets a new performance standard at conventional static memory prices. When accessed through I/O port channels, the half megabyte V-RAM becomes M Drive compatible with true non-volatile solid-state disk capability.



MI-286.

ADIT.

MSR. High performance and reliability in a memory so fast you won't believe it's a dynamic ram product. Compatible with all popular S-100 environments, the MSR's low power consumption and 120 nanosecond ram devices set a new standard for dynamic memory products. The MSR is available in quarter, half, one and two megabyte configurations at the lowest prices in the industry.



V-RAM.

MSR.

Dealers:

Vision Computers	(415) 530-2390
Compliance Computer Technology	(800) 222-8686
S-100	(800) 423-5922
John D. Owens & Associates	(212) 448-6298
In England; Fulcrum (Europe) Ltd.	(0621) 828763

Macrotech dealers also include most Compupro Systems Centers, Heathkit Electronic Centers and Alpha Micro Dealers.

MACROTECH International Corp.
21018 Osborne Street, Unit #5
Canoga Park, CA 91304
(800) 824-3181 • in Calif. (818) 700-1501
Telex: 9109970653

editor interface

Editor Interface *provides a means for readers to share their comments and questions with us. Letters are published on subjects concerning the magazine, any of the articles, or any other topic of interest to the S-100 community. Please send letters to Editor Interface, S-100 Journal, PO Box 1914, Orem, UT 84057. If possible, please type (print) your letters.*

Because of time and space constraints, I am not able to answer all the letters received, but I do send out several personal replies in addition to those published here. Whether or not answered, I read all the letters and use them in deciding what articles to publish in the magazine.

Keep those comments coming, and thank you for your support of S-100 Journal.

S-100 Support

CCT is undoubtedly the largest and oldest S-100 OEM in existence. We have been installing S-100 based machines since 1976. Over the years, we have budgeted many hundreds of thousands of dollars to advertising, which we have always felt an investment, rather than an overhead expense.

Obviously, since the new consumerism of the PC, most other publications have swung to cater to this audience. CCT does not, nor has ever, dealt with any PCs of any kind. We feel these are nothing but consumer products of inferior design and construction, aimed at the inexperienced neophyte. CCT has built its reputation and tremendous user base by providing powerful and reliable systems on the S-100 bus. As a result of this, CCT has not advertised in a commercial publication for almost 10 months.

I am pleased to finally hook-up with you, and further, enjoy your strictly S-100 attitude for the magazine. Enclosed is the ad layout for the Winter issue. I welcome the chance to again advertise to promote CCT products and services.

I would also be willing to contribute to the magazine much in the way of technical data, bug fixes, and general information addressing Digital Research operating systems and CompuPro and Macrotech based hardware.

Pat Martini
President
CCT
Prescott, Arizona

Most everyone who has kept up with the S-100 community for several years is aware of CCT's longstanding dedication to the S-100 bus. I am very pleased to welcome you to S-100 Journal. I very much admire your strong commitment to the bus in these days when many systems houses that grew by selling quality S-100 components have sold their reputation for a cheap buck.

I look forward to a long and mutually supporting relationship with you and your company. •*Jay*

OS Coverage

I have enjoyed *S-100 Journal* a great deal and look forward to each issue. I would like to see expanded coverage of multiprocessor hardware, OS's, and system programming techniques. I primarily work with TurboDOS, and would like more coverage of it, but I would also like to hear of other comparable OS's. I would also like to hear from anyone working with or developing a fault-tolerant S-100 system. Keep up the good work.

Chris Wallis
Eureka, California

As S-100 Journal grows, one of the plans that we have is to expand the Multiuser OS column to include several articles and authors in each issue. The column is now in sort of an introductory phase for each operating system, but it will become more practical, interactive, and useful. •*Jay*

Schematics Needed

Glad to see your magazine is available, I'm sending in my subscription. Do you know where I can get schematics/data on Computime clock/calendar T102A; Micromation disk controller; and/or Solid State Music I/O (2 serial, 2 parallel)? Thanks.

Fred Ordway
2901 Telestar Ct.
Falls Church, VA 22042

I have no information about the first two boards, but I've published your address in case another reader knows. I do however have an old number for Micromation that still rings, but no one answered when I called: (415) 398-0289. For the schematics of the SSM IO-4 board, which happens to be pictured on page 23, contact Brett Clippard at Transend Corporation, 884 Portola Rd., Portola Valley, CA 94025, (415) 851-3402.

Thanks for the sub. •*Jay*

S-100 Primer

Are there any basic primers on the S-100 bus for beginners?

Any information on getting started, or where to find the information, would be appreciated.

Thomas Kent
Salem, Oregon

Unfortunately, there are no primers that I know of. It is possible that we will publish one in the near future. Presently, your best source of information is to continue reading S-100

(continued on page 52)

SIMPLE INTERFACING TECHNIQUES

Don Pannell

Do you have a technical question about the S-100 bus?

Is there an area of the IEEE-696 standard that you do not understand?

Do you need to know how to interface a particular device or function to the bus?

696 Bus *is our regular column that helps clarify questions and concentrates on the hardware aspects of the IEEE-696 bus (i.e., the S-100 bus).*

Don Pannell, our S-100 bus expert, has been using S-100 systems and designing S-100 boards since 1978. Don is a coauthor of the IEEE-696 standard. This is the standard that defines the rules of operation of the S-100 bus. Adherence to the standard permits boards from different companies to work with each other.

If you have a question about the S-100 bus and IEEE standard, please type your question and send to Don Pannell, PO Box 700112, San Jose, CA 95170-0112.

In future columns and as space permits, Don will incorporate answers to questions received from readers.

More than once you may have desired an additional special function or feature on your computer. An extra parallel or serial port, maybe? You may even have thought of building what you needed on your own. It isn't hard. With the S-100 bus, designing an Input/Output (I/O) card is probably easier than you think. I will show you an example of a parallel port and explain some of the DOs and DON'Ts of S-100 bus designing.

A QUICK OVERVIEW

The S-100 bus is a set of 100 electrical signals used to transfer information from one place to another. To simplify this quantity of signals, let's group them according to the age-old questions any good detective asks when encountering an unknown:

WHO Is in Control?

The **BUS MASTER** controls the bus. Normally this is your CPU card, but a DMA (Direct Memory Access) device can control the bus as a Temporary Bus Master (this is a topic for a future issue). The important piece of information to remember is that **all** signals are referenced to the BUS MASTER (see next paragraph).

WHAT Is the BUS MASTER Doing?

The BUS MASTER is always controlling the transfer of information (**data**). About 90% of this information is the program your CPU is executing. Of the remainder, approximately 8% is the data the program is manipulating — about 4% being data reads and 4% data writes. The final 2% is I/O. In the simplest case, 8 bits (a **byte**) of information are transferred at a time. This transfer takes place through one of the S-100 bus' two unidirectional 8-bit data paths. One data path is used for the **output data** while the other is used for the **input data** (data direction is always referenced to the BUS MASTER). The S-100 bus supports 16-bit data transfers as well. Refer to Kevin Parker's article on page 38 for information on this advanced topic.

WHERE Is the BUS MASTER Transferring the Data?

The **address lines** of the S-100 bus specify **where** any transfer is to take place. The most common places are memory and I/O boards. With the 24 memory address lines supported by the IEEE-696 standard, up to 16 Megabytes (2^{24} bytes) of information can be directly available in any system. The S-100 bus also supports up to 64K (2^{16} bytes) of I/O-device addresses by sharing the lower 16 address lines for I/O transfers.

HOW Does the BUS MASTER Specify What it Wants to Do?

It indicates its desire on the 8 **status lines**. The status lines are one of the strong features of the S-100 bus. They are used by the BUS MASTER to inform the rest of the system what it is going to do **before** it does it.

WHEN Does the Transfer of Information Take Place?

Only during the bus **read** or **write strobe**. The BUS MASTER controls these strobes, and it is the responsibility of the **selected** device (the card you build) to transfer the data only during their assertion.

These are not the only groups of signals supported by the S-100 bus. But they are the only ones needed in order to build a good-quality I/O card. This is not to say the other signals are unnecessary; just that a majority of I/O cards do not require the features gained by using the other signals.

WHERE TO START

First, you need to decide how many I/O addresses the card will require (the number of addresses will depend on the on-board features), and what I/O-addressing scheme you are going to use. This is not a trivial question. There are two addressing schemes, and they are not necessarily compatible. The first scheme uses only the lower 8 address lines for a total of 256 different I/O addresses. The second uses the lower 16 address lines for 65,536 (64K) I/O addresses. Very few cards use the extended 16-bit I/O address scheme. This is mainly due to cost and the desire for a board to be usable with the older 8-bit CPUs that can only support a maximum of 256 I/O devices.

I recommend using the 256-I/O-address mode for compatibility and simplicity. This mode may only be a problem if your card needs more than 32 I/O addresses. A maximum of 32 I/O addresses allotted to each I/O card allows eight I/O cards in the system. However, I like to limit the maximum number of I/O addresses for any one card to 16. This gives room for growth, in case more cards are needed later.

After the I/O-address scheme has been decided, the board-select logic can be designed (Figure 1). The board-select logic allows the card to recognize when the BUS MASTER is addressing it. It is important to remember that we want this card to respond only during I/O bus cycles directed to it. Therefore it is essential that we can determine when an I/O cycle is taking place.

An I/O cycle is defined by either sOUT or sINP being high. sOUT and sINP are two of the eight status signals available on the S-100 bus. sOUT is high only during I/O output cycles, and sINP is high only during I/O input cycles (these signals are *active high*, meaning that they are TRUE when high — at or over 2.4 volts). A logical OR'ing of these two signals (using three gates of a 74LS00 chip — see Figure 1) is combined with the four address lines A4 to A7 to produce the active-high, on-board signal BRDSEL (board select). A 4-position DIP switch is used to set the I/O address range for the card. The 74LS85 chip only activates the BRDSEL signal when the signals on address lines A4-A7 match those preset on the four lines coming from the DIP switch and either sOUT or sINP are high.

NOTE: Unused TTL inputs must **never** be left floating (unconnected). This is the reason for the pull-up resistors on each of the DIP switches' outputs. These resistors insure a high input to the 74LS85 whenever the switch is open.

BUFFERING BUS SIGNALS

The lower four address lines (A0-A3) are buffered with one half of a 74LS244 on their way to two 3-to-8 decoders (74LS138s). One decoder is used to generate eight active-low read strobes and the other generates eight active-low write strobes. Address line A3 must be low for these decoders to be active (due to A3's connection to one of the low enables of the 74LS138s). The important item to remember here is the buffering of the address lines. The 74LS244 adds important features. These are:

1. Insures that the S-100 bus sees no more than one 74LS-TTL input load. Actually a 74LS244 is one half of a 74LS-TTL load which is even better.

2. Proper placement of the 74LS244 close to the S-100 bus allows the shortest possible distance for the wires coming from the bus. Short wires from the bus prevent signal noise in high-speed systems. This permits the card to work in any

IEEE-696 STANDARD UPDATE

CompuPro has been granted a Project Authorization Request (P.A.R.) from the IEEE for an enhancement to the IEEE-696 bus standard. The only enhancement covered is an allowance to supply regulated as opposed to unregulated voltages on the bus. All boards must have the ability to accept on-board voltage regulators for compatibility with older systems. Furthermore, boards without the regulators installed must be clearly marked for use in these special systems.

A draft of the proposed changes has been mailed to all previous committee members for their vote. The votes were due back by December 19th, 1986. As of mid January, 1987, all but three of the past committee members had responded and only 5 'no' votes were received. The 'no' votes will most likely be changed to 'yes' votes once their objections to the draft are removed. The modified and hopefully final draft should be out to the members by the end of January and the change presented to IEEE in the month of March.

Richard L. Kalish, Director of Hardware Development at CompuPro, is chairman of the IEEE working group for the voltage specification change. He is also chairman of the study group for future enhancements to the IEEE-696 standard. These enhancements include a 32-bit data bus (please see, in this issue, the article by David Plomgren on a 32-bit expansion to the S-100 bus).

Don Pannell

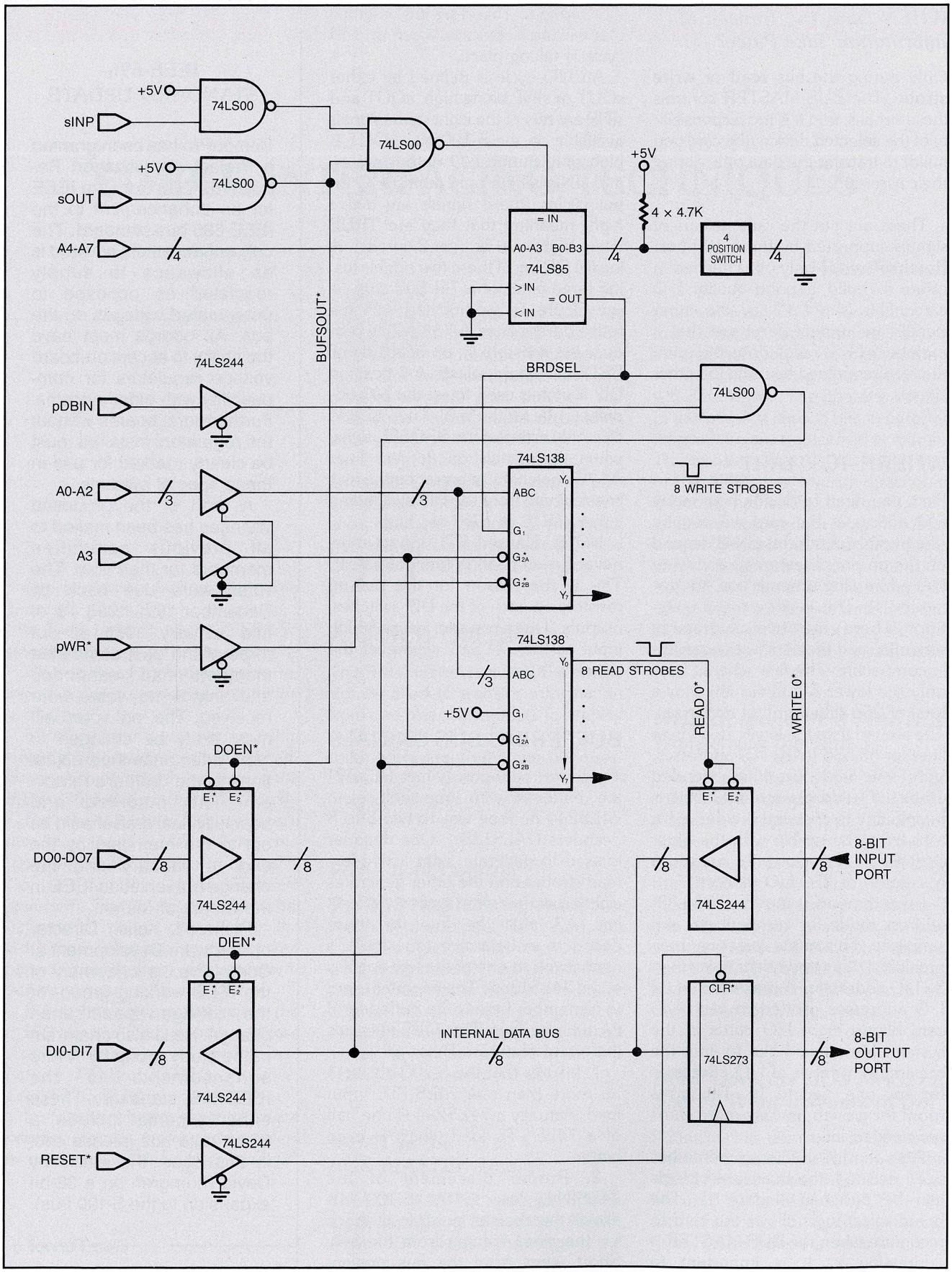


Figure 1. *A circuit design to interface a parallel port to the S-100 bus.*

slot of the system (no more dependencies on card position).

3. A 74LS244 is a 'schmitt triggered' device which means it sharpens up slow-edge speed signals. This helps the new card ignore any bus signal noise that may be present in the system.

4. The buffering of these bus signals allows future expansion of the card without the need to rewire the first section of the card. One possible expansion would be to add two more 74LS138s for eight more read and write strobes enabled when A3 is high.

The two strobe signals, pDBIN and pWR*, and the bus reset signal, RESET*, are buffered with a 74LS244 mostly for the same reasons stated above. The other half of the 74LS244 used for buffering the address lines can be used here (there are eight buffers in a package).

Buffering the bus strobes is important. Not only do these signals control the actual transfer of data, but these are the narrowest, or shortest-width, signals on the bus. Bad bus strobes lead to improper clocking of data.

The RESET* line is also buffered for another reason. It is most likely the widest, least switched signal on the bus. However, it tends to be connected to a lot of physical devices. I have seen some systems with their reset nets so long and heavily loaded that occasionally 'out of the blue' the system would get reset or partially reset due to noise on its reset net. As with any noise related problem, this type of bug is very hard to find.

DATA BUS CONTROL

The S-100's two 8-bit unidirection buses (DI and DO) are combined on-card into an 8-bit bidirectional bus and at the same time buffered with two 74LS244s. Proper control over these 74LS244s is very critical.

The data-in bus, DI0-DI7 (data in to the BUS MASTER — data out of this card), is enabled only on an I/O read taking place to this card during the read strobe (pDBIN). The remaining 74LS00 gate is used to combine the board select (BRDSEL) with the read strobe. The read strobe must be used to enable the data-output drivers because the I/O-read status signal (sINP) may go 'active' during bus transfer cycles (i.e., DMA) causing data bus contention.

The data-out bus, DO0-DO7, is enabled into the card on any I/O output by means of an inverted copy of the status signal sOUT. This allows liberal data-setup and hold times on both sides of the write strobe (pWR*). It also insures that the data-out buffer is disabled during I/O reads.

THE PARALLEL PORT

Only one parallel port is shown on the schematic (Figure 1). The output section consists of a 74LS273 connected to the on-board Write Strobe 0. Write Strobe 0 will only be activated when this card is selected (BRDSEL is high), the lower four address lines are 0, and the bus write strobe (pWR*) is on. This is all handled by the three 'enable inputs' to the 74LS138.

The input section of the parallel port is a 74LS244 connected to Read Strobe 0. Read Strobe 0 is controlled very similarly to Write Strobe 0 except that the gated version of the bus read strobe (DIEN*) is used. This requires the unused input to the 74LS138 to be tied high.

NOTE: It is legal to tie unused inputs that need to be 'high' directly to +5 Volts for the TTL families 74LS, 74F, 74ALS, and 74AS. Regular TTL, 74H, and 74S all require that a current-limiting resistor be used. An easy way to tell if the resistor is required is by looking at the device's absolute maximum ratings for Vcc and its inputs. If both have the same limit and go to around +7 Volts, the current limiting resistor is **not** required.

A second parallel port identical to the first can be added by using any one of the other eight read or write strobes in the two 74LS138s. Up to eight parallel ports could be added.

POWER SUPPLY CONSIDERATIONS

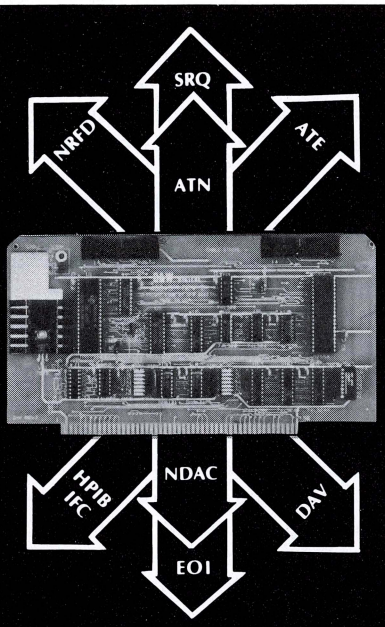
Just about any +5-Volt regulator that can supply sufficient current for all the components will work. Most cards will require the +5-Volt regulator to be on a heat sink. This is a good idea even if the initial power requirements don't warrant the use of one. Because of the heat sink, when you later add more functions to the card, you won't have to rewire the power regulator section.

If you are wire wrapping a card without a power grid on it, make sure that you fabricate a power grid when connecting the IC's. Don't worry about 'ground loops' on such a small card. I recommend that each IC derive power and ground from two points. This insures that all the parts get sufficient power even if a bad wire or wire wrap exists. Connect all the IC's power and grounds first, before doing any signal wires. Doing so keeps the power connections on the bottom of all the wires, helping to make certain they are not disturbed during debug or rework. Using red wire for power, black wire for ground, and blue wire for signals also helps. Good power connections are essential for reliable cards. Any extra time and care spent here is saved tenfold later.

Last but not least, don't forget the filter capacitors. You should have a 50 to 100 uF (micro Farad) electrolytic cap before the +5-Volt regulator (on the +8-Volt line) and one after the regulator (on the +5-Volt line). Three to six electrolytic caps in the range of 4 to 10 uF should be distributed around the board as well, and about every other IC should have a 0.1 uF cap between +5 Volts and ground. Lack of these filter caps results in cards with signal noise problems, which can be extremely difficult to debug.

OTHER DESIGN POSSIBILITIES

Not all systems need a parallel port like I have described here. Your system may need a serial port or a floppy disk interface. Unfortunately, these other designs are a bit more complicated because you are most likely connecting a specialized chip to the S-100 bus. When attempting one of these designs, pay particular attention to the chip's specifications on data setup and hold time, and read and write strobe widths. To increase the strobe widths, you may need to add a wait state generator to the card. In my last



THE 488+3

IEEE 488 TO S-100 INTERFACE

S-100 ⟺ 488

- *Controls IEEE 488 (HP1B) Instruments with an S-100 computer*
- *Acts as controller or device*
- *Basic and assembly language drivers supplied*
- *Meets IEEE 696 specification*
- *Industrial quality burned in and tested up to 125K bytes/sec under software control 3 parallel ports (8255-5)*
- *$375*

D&W DIGITAL, INC.
20655 Hathaway Avenue
Hayward, California 94541
(415) 887-5711

GOOD BUILDING PRACTICES

1. Route a good power-supply grid prior to any signal wires (use red and black wires).
2. Provide sufficient filter capacitors on your power grid.
3. Never load any S-100 bus signal with more than one 74LS-TTL load (two to three 74ALS-TTL loads are OK if the signal wire is very short).
4. Keep all S-100 bus signal wires shorter than 2 inches where possible. This is extremely important for address, data, status, and strobes.
5. Never use any of the S-100 bus' status signals to clock data into a device. Use the bus strobes (pDBIN and pWR*) only. The status signals cannot be used because they are undefined and can appear to go 'active' during bus transfer cycles (i.e., DMA).
6. Never leave an unused TTL input unconnected.

column (S-100 Journal, Vol. 1 No. 4), I described how to do this. The same issue also contains an excellent article by Kevin Parker on a serial port design.

NOTES FOR MEMORY CARDS

Memory card designs require a few added comments. First, decoding of the S-100 status lines is not as simple as it is for I/O cards. Do not forget the special case of Interrupt-Acknowledge cycles; they look almost identical to a Memory-Read cycle. You don't want your memory card responding as the interrupt vector data. Second, a good memory card needs to support the PHANTOM* signal and disable itself whenever PHANTOM* is asserted. And third, memory cards need to contend with a third bus strobe called MWRT for memory write.

Refer to the '1-Megabyte Memory' article on page 38 for more information on memory designs.

CONCLUSION

As you can see, I divided the S-100 bus lines into groups that answer the age-old detective's questions of WHO, WHAT, WHERE, HOW, and WHEN. Answering these questions while you are designing a board should ease the design process.

I have also outlined some good building practices when interfacing to the S-100 bus. They are summarized on the above table. ■

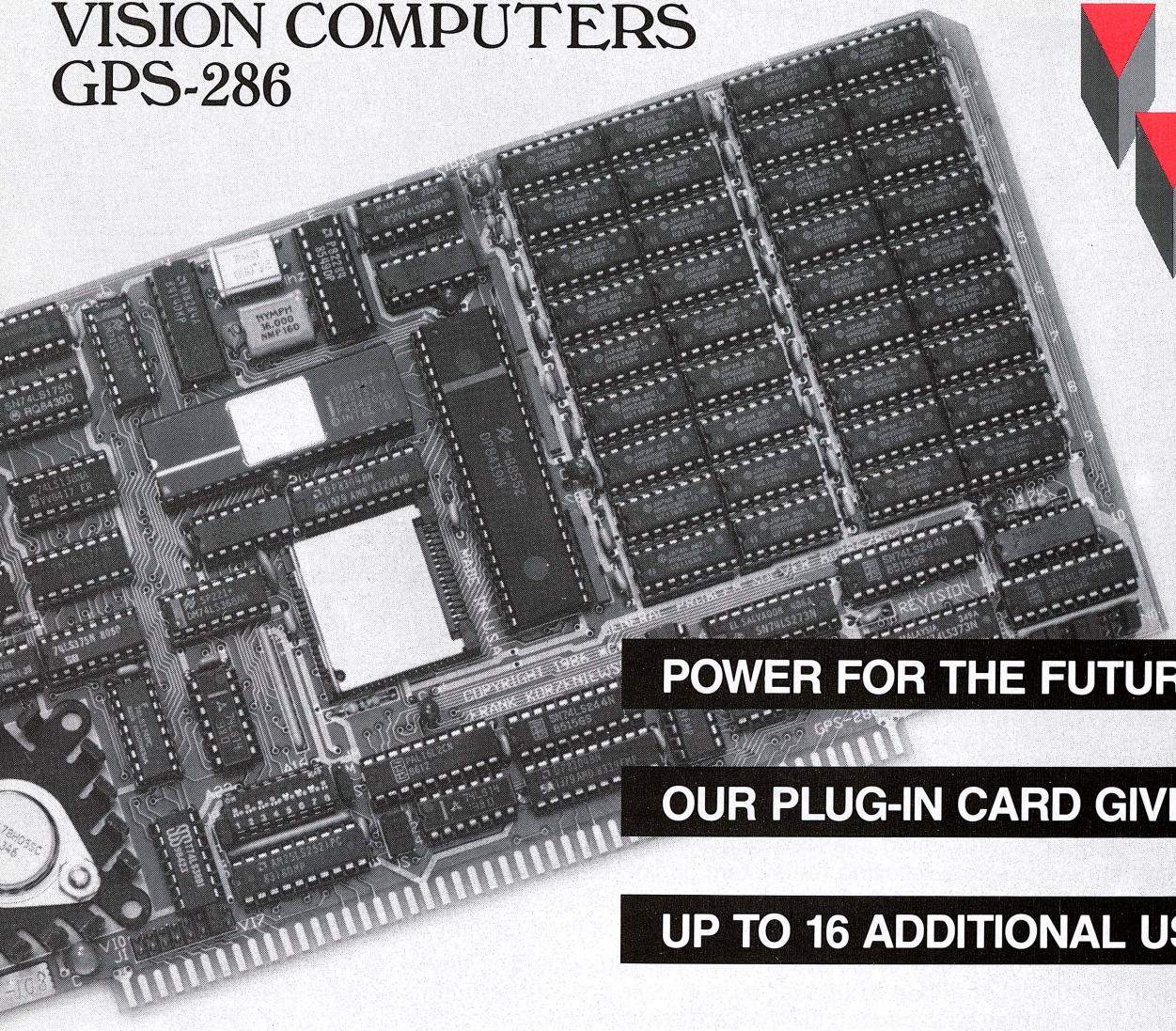
ERRATA

Some errors have accidentally been printed in past 696 Bus *columns. They should be corrected as follows:*

Vol. 1, No. 3, page 35 states 'tRDYΦ, setup time RDY, XRDY, SIXTN* to Φ rising, was defined to have a minimum time of 70ns. The minimum time is now 20ns.' *It should read:* '...was defined to have a minimum time of 80ns. The minimum time is now 70ns.'

Vol. 1, No. 4, page 12: The middle column of Table 4 lists some delay numbers for D1=ALS175. *These delay times for an* ALS175 *are in error and should be the same as the numbers listed for* D1=LS175. *The fastest times would be obtained by* D1=AS175 *and these numbers are* 42ns, 43ns, and 20ns (for B1=LS38, ALS38 and S38 respectively).

VISION COMPUTERS
GPS-286



POWER FOR THE FUTURE ...

OUR PLUG-IN CARD GIVES YOU

UP TO 16 ADDITIONAL USERS!

Mainframe Power!

Presenting our GPS-286 package which can quickly turn your CompuPro S-100 computer system into a million dollar mainframe. With a 8 MHZ 80286 and a megabyte of RAM, the GPS-286 opens up a whole new world of business computing that is unconstrained by CPU speed or memory capacity. It keeps you working instead of waiting!

Why You Need GPS-286

The GPS-286 was designed for more users and faster performance. A single user can take advantage of a megabyte of RAM along with a dedicated, no-wait, 10 MHZ 80286, or up to 16 users can share. The GPS-286 satellite is a master at efficient resource allocation, and adapts instantly to your changing system requirements. All users connected to the system share floppies, hard disks, and solid state disks, as well as printers, and modems — without the overhead of a network. This means ultra high performance and reliability at a rock bottom price.

Fits Your Applications

Just imagine 15 users running WordStar or NewWord, or 11 users using Medical Manager or MC Accounting software, or how about 6 Dataflex applications. All at the same time. Need more applications? Choose from any of the hundreds of other Concurrent DOS applications. Add a second or third GPS-286 and you've got 12 or 18 Dataflex users! Since you can have up to eight cards in a system, the skys the limit. That means lots of applications for a lot of users.

Easy Upgrade

You can install the GPS-286 in ten minutes. No new operating system, no new commands and no complications of a network. Both the CompuPro and Gifford operating systems are supported, including external networks to other CompuPro based systems and IBM PCs, XTs, ATs and compatibles.

Your Choice

With up to 64 MHZ of 80286 processing power! and 8 megabytes of RAM for user applications! Do you want mainframe performance?

Call Now

Call Vision Computers to find out how you can experience the future today at (415) 530-2390.

Vision Computers
2235 Melvin Road
Oakland, CA 94602
Telex: 910-240-2681

Compliance Computer Technology, Inc.

CCT®

A PAT MARTINI COMPANY

CCT Bldg • 6476 Airpark Drive • Prescott, AZ 86301

For information and technical support: 602-776-1188

For ordering and systems quotations: 800-222-8686

CCT will provide FREE technical help to anyone — Call!

- *CCT is one of the oldest and largest S-100 OEM's in the world. Our large user base is comprised mainly of CompuPro and Macrotech based multi-user systems. We have produced thousands of S-100 systems since 1976.*

- *CCT manufactures over 30 disk drive subsystems for S-100 machines, including 3 inch, 5 inch, and 8 inch floppies, and hard disks from 10 to 300 megabytes. These subsystems are designed and manufactured to the industrial quality requirements of our large governmental and large business user base.*

- *CCT sells, services, and supports the complete lines of both Macrotech and CompuPro products. We can provide turn-key packages to upgrade your present system using state-of-the-art hardware and the latest generation operating system releases.*

Limited Special for all S-100 users: The CCT PRINTERFACER 1

The Printerfacer 1 is a self contained standard IEEE-696 S-100 board which is both an output buffer and device interface. The board contains a Z80 processor, serial and parallel I/O ports, operating firmware, and up to 1 megabyte of on-board memory. The Printerfacer 1 accepts output from the system at bus speed, while buffering characters and maintaining interface with a parallel and a serial printer and/or plotter. This speeds up throughput tremendously by relieving the system of slow handshaking protocol. With a simple command the Printerfacer 1 output can be toggled between the parallel and serial devices, thus allowing accessability to both at the same location, at the same time, without any software or operating system changes!

The Printerfacer 1 will work in any S-100 system, in a polled or interrupt driven environment, and also has a special CompuPro emulation mode to appear as any user port of the Interfacer 3 or 4. The board comes standard with 256K and may be expanded to 1 megabyte by simply plugging-in standard 256K bit DRAM chips. The Printerfacer 1 tests memory and self-sizes at each cold start.

CCT will configure and test the board for your specific system requirements, and can also furnish information on addressing as the primary LST device or at another device location. Plug-in lightning output speed and double your device capability.

Limited Time Special: $249.00 plus shipping.

For Concurrent DOS users: CCT CMX Concurrent DOS operating system

The CCT CMX Concurrent DOS operating system is written for end user flexibility and speed. The CMX 2.1 release supports any floppy and hard disk combination. The CMX XIOS is designed to include drivers, tables, and buffer space for only those devices actually in use in the particular system. The end result: A more compact, efficient, and faster environment in less memory space. Run CP/M 8 bit and 16 bit software, and MS-DOS applications under the most flexible micro operating system today. Upgrades available to CompuPro licensed users.

CCT Implements Tomorrow's Technology Today!™

CCT, CMX, Printerfacer 1, and CCT Implements Tomorrow's Technology Today! are Trademarks of CCT. CP/M and Concurrent DOS are Trademarks of Digital Research. MS-DOS is a Trademark of Microsoft. Interfacer 3 and Interfacer 4 are Trademarks of Viasyn Corporation.

INTRODUCTION TO TurboDOS

George Carmack

Multiuser OS *is a regular column that discusses multiuser operating systems currently running on S-100 microcomputers. The column discusses such operating systems as Concurrent DOS, TurboDOS, Unix, d/OS, AMOS/L, OS-9, THEOS, etc.*

Each issue features one or more operating systems in articles written by one of a group of columnists who are experts on the particular OS they write about.

Readers are invited to submit questions they may have about any of the operating systems discussed. Answers to questions will be incorporated into future issues as space permits. Please send any questions or comments about this column to Multiuser OS, *S-100 Journal, PO Box 1914, Orem, UT 84057. Letters should be typewritten, and each letter must deal only with one OS in order to facilitate forwarding to the proper columnist.*

We still have available a few columnist positions for Multiuser OS. *If you have a solid knowledge about a multiuser operating system, you have access to a version of that OS running on an S-100, and you would like to become a columnist, please contact S-100 Journal.*

In this issue, the operating system featured is TurboDOS. George Carmack is president of Microvations, a California systems house that specializes in business S-100 systems running TurboDOS.

This article introduces TurboDOS to those not familiar with it and discusses its features and offerings. TurboDOS may be the least publicized of operating systems. Its authors have left promotion to those companies manufacturing products designed to use it and, in spite of the fact that some fifty to sixty thousand site licenses have been sold, TurboDOS is still relatively unknown. Among the licensees are universities, defense contractors, Fortune-500 companies, our government and military, and 'the corner grocery store.' Represented are systems from 1 to more than 100 users.

CONNECTIVITY

Without discussing the strengths of individual chips, we can all agree that there is a great deal of power present in today's microprocessors. The idea of expanding this power by joining the processors together into some collective, working unit certainly has large appeal. A major theme at Comdex Fall'86 was **connectivity**. We have seen the recent success of such companies as Novell and Corvus in joining PCs, and certainly there is more joining ahead. But 'joining' is something TurboDOS has been doing since the early 80s. Today, it's a proven and reliable operating system.

TurboDOS is a network operating system. There are many buzzwords used today in association with *network*, so let's simplify things by calling this a collection of processors sharing 'peripherals.' These 'peripherals' are commonly disks, printers, and other devices. The way in which the processors are connected (the collection method) is usually the S-100 bus or a cable system. The connection is called a circuit. On each circuit are found nodes. Nodes are generally the user processors, and we commonly refer to them as *slaves*. There must be a circuit controller, called the *master* or *file server*. The master manages the information flow to and from the slaves and may be part of more than one circuit. Master to master linking (multiple circuits) allows very large networks to be constructed. Any shared peripherals are controlled by the master in a given network. This is, of course, a simplified explanation of a TurboDOS network. A more in-depth coverage was published in S-100 Journal, Vol. 1 No. 4 in an article by Akin Orhun.

WHAT TurboDOS OFFERS THE USER

TurboDOS offers primarily a CP/M-80, CP/M-86, and MS-DOS software interface. There is still a large body of software available for CP/M-80, although new development is sparse. There are one or more versions of nearly every computer language, and many quality public domain utilities and mature commercial products. CP/M-86 lost the 16-bit OS battle early on, and software here is rather weak, but the amount available may surprise you. Most of the 8-bit languages came over to CP/M-86 and were followed by the corresponding commercial packages. While new development is very scarce, there are still quality packages being sold. In the MS-DOS side of

things is where most of today's software work is being done, and the selection and variety is huge. Altogether, this boils down to nearly all the significant microcomputer software written in the last ten years. The MS-DOS interface to TurboDOS is commonly provided either by S-100 bus slaves running an emulation or the direct connection of a PC-type machine by high-speed link.

Access to Common Peripherals

Access to peripherals can be controlled in many ways to allow or deny the use of a device to any given workstation. Users may be restricted to the use of certain disk drives, printers, and file areas. This gives a system integrator great latitude in the setup and placement of peripherals in a given network.

Spooled and Despooled Printing

TurboDOS can handle up to sixteen printers and offers a number of ways to control print routing. Under most conditions, printing is spooled to a disk file and then despooled to the assigned printer as a background task. This is transparent to the application software which quite happily assumes that it is conversing with a real printer. No block of RAM is dedicated to this process, and the creation and deletion of the spool files is performed automatically by the operating system. Utilities are provided to control printing from the command-line level. Control is also provided through operating system function calls.

Logon Password Security

Not only may a workstation be given or restricted access to specific resources, but a user's activities can be controlled and logged. The work area allocated to a user and even the program he or she runs are configurable. System security is a large concern to many organizations and is a solid part of TurboDOS.

A Simple, Powerful Command Line Interface

TurboDOS contains a command line interpreter which presents the familiar A> prompt. Included are utilities for the management of the system such as copy, rename, file attribute setting, directory listing, printer assignment, batch processing, etc. Auto-executing 'shells' are possible; they shield the command line interface from those for whom it is not necessary.

WHAT TurboDOS OFFERS THE PROGRAMMER AND THE SYSTEM INTEGRATOR

TurboDOS provides CP/M and MS-DOS compatible file and record locking. Any multiuser system must have ways to protect data files from record-update collisions. In Digital Research's MP/M operating system, file and record locking capabilities were present; they were utilized in several language compilers and, subsquently, by commercial software companies in products developed on those compilers. TurboDOS provides compatibility with the DRI locking scheme. DRI's Cbasic compilers (and perhaps some of their other compilers) and Ryan/McFarland COBOL are examples of languages which handle record locking properly under TurboDOS. Database systems such as DataFlex and Q-PRO4 also work well. TurboDOS record locking is also compatible with version 3.X of MS-DOS.

Wide Choice of Languages

Versions of almost every computer language ever used can be run on a TurboDOS system. Some are available for all three compatible operating system interfaces. For example, one might use a C compiler with versions for MS-DOS, CP/M-86, and CP/M-80, addressing the broadest possible market. The migration of software from 8-bit to 16-bit systems allows for growth in performance and function without total obsolescence of the computer system.

Extensions to the Operating System

TurboDOS also provides a means for a programmer to create functions and link them to the operating system itself. In the past, this has been used for such tasks as bulletin board or modem support routines, network mailbox systems, and directing a console's output to several workstations for training purposes. While extending the operating system is not necessarily the simplest thing to do, it can and has made a great contribution to the usefulness of a system. Other examples in use today are real-time data monitoring systems and time-scheduling of job execution.

Easy Access to Operating System Functions

The normal OS function calls are supported. In addition, there is an extended set of functions reaching the special abilities of TurboDOS. Operations such as reading the date and time, redirecting printer output, and setting communication channel characteristics are available to a programmer. Many languages have support for operating system function calls, making it relatively easy to use all the features of TurboDOS.

Large 8-bit Program Area

Most of today's 8-bit slave boards utilize the banked memory feature of TurboDOS, providing a program area of 63.5K. While this seems small by 16-bit standards, it enables larger program modules. It has really helped in keeping alive the use and functionality of 8-bit components in systems. There are still many computer applications that are well suited for 8-bit processors, and it is not necessarily correct to say they are slower. The 8-bit languages are generally very efficient and well tuned to the processor and memory environment.

Large Disk and File Sizes

TurboDOS supports disk drives up to more than 1000 Megabytes and files of up to 134 Megabytes. This allows

the use of today's largest and fastest drives and makes it possible to perform very large data processing tasks. The organization of data on a TurboDOS disk yields extremely fast access to files on an unambiguous basis.

Interprocessor Communications

Here lies my favorite feature and part of what multiprocessor systems are all about. If there is a simple method of conversing among processors, why not set up certain processors to perform only specific tasks. For example, we have all had to wait at our keyboards while a report was running, unable to perform any other work. Print spooling certainly helps here, but you are still grounded while the report generates. By dedicating one processor to handle all reports (or any program resulting in loss of keyboard), a programmer can dramatically increase the productivity of the available workstations. If you can stand another buzzword, this is called **distributed processing** and can be done very effectively under TurboDOS. The method most often used is a special file called a FIFO file. Normal data files can be read and written either sequentially or by random record number. In a FIFO, data is always read from the file's beginning and written to its end. This gives a 'First In / First Out' flow of information that lends itself nicely to setting up job queues and message packets in an orderly and predictable manner. FIFO files are typically used for such tasks as massaging the data collected in real time by one or more other processors and rapid credit card verifications by modem during on-line order entry. They are also used in support of an index managing processor. This processor contains all the index management code for the entire network and performs lookups at the request of other processors in the system. The code in the other processors is greatly reduced, and index integrity is extremely high.

OPERATING SYSTEM STABILITY

At this point in time, TurboDOS is a very mature and stable product. This is also true of much of the S-100 hardware available for use under the OS. With stability comes reliability, and there is certainly much to be said for that. A properly designed and constructed TurboDOS system represents a large amount of computing power for its cost, with easy and economical growth and expandability.

Also, microcomputer networks are a major part of computing today and will surely grow in scope and influence. TurboDOS helped pioneer some of this ground and is worth a solid look from anyone contemplating buying or selling microcomputer networks. ■

Z80 Turbo Modula-2

Created by Borland International!

Complete programming environment features WordStar-like editor, automatic invocation of editor when compiler detects programming errors, and extensive function library

Turbo Pascal programmers will love it because: separate compilation for large programs makes development fast, advanced capabilities for linking assembly language REL and object files, passable open array parameters, and much more

Easy to use: includes extensions like Turbo Pascal for I/O, string handling, multidimensional arrays, and error handling, makes use of Modula-2 much less complicated

Includes multiprogramming facilities for advanced applications, I/O redirection to and from files and devices

Outperforms "C" compilers for speed of compilation and speed of execution

500-plus page Turbo Modula-2 manual included

Price: $69.95

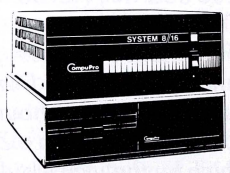
Requires Z80-compatible microprocessor and CP/M compatible operating system

Dealer and Distributor inquiries invited.
Visa/Mastercard accepted. Add $6.00 shipping/handling in North America, actual cost elsewhere. Specify disk format.
Trademarks: Turbo Modula-2, Turbo Pascal, Borland International; WordStar, MicroPro; CP/M, Digital Research; Z80, Zilog.

Echelon, Inc.
885 N. San Antonio Road • Los Altos, CA 94022
415/948-3820 (Order line and tech support)

CompuPro™

SYSTEM UPGRADES



- More Users
- New Operating System
- Larger, Faster Hard Disk
- PCs Networked to System
- Custom Database Software

Get it done right the first time

(415) 495-5999

COCHLIN COMPUTER SYSTEMS

330 TOWNSEND STREET #107 SAN FRANCISCO, CA 94107

UPGRADING OLD ALTAIRS

Daniel E. Zabriskie

Here we are again bringing the Homebrewing *column for those S-100'ers of character who are in love with front panel lights and switches.*

Homebrewing *is our fun column entirely dedicated to sharing information about hacking with S-100 systems. If you have done any interesting projects, resolved nagging hardware problems, or turned your system inside out with your creativity, do share your experiences with us. Send your typed double-spaced contributions, either letters or complete articles, to* Homebrewing, *S-100 Journal, PO Box 1914, Orem, UT 84057.*

For those still using an original unmodified Altair, Daniel Zabriskie shows in this issue how you can run it with CP/M or with a Z80 CPU card (which by the way would allow running Echelon's Z-System for the ultimate in 8-bit performance).

Daniel Zabriskie, a senior field engineer for Computer Curriculum Corporation in New York, also runs his own S-100 systems house specializing in inventory and accounting solutions for local businesses. He can be contacted at Colonie Computech Corp., 438 New Karner Road, Albany, NY 12205-3810.

The Altair 8080 computer can be easily modified to use a Z80 CPU and to run standard CP/M-80 (CP/M-2.2). Either or both modifications may be performed, depending on your budget and needs. The Z80 upgrade will double the processor speed, while allowing you to continue using your Altair disk-based programs. The CP/M conversion will allow you to run 8080 CP/M, and its zillions of available programs, on standard soft-sectored disks using your Altair hard-sectored drives and either the Altair 8080 CPU or the Z80.

CONVERTING THE ALTAIR TO CP/M

The Altair S-100 computer can be converted to run standard CP/M-80 with relative ease. The simplest method involves replacing only the disk controller. I used a Tarbell double-density controller, the DD-FDI, Rev. F. I even was able to use the Altair floppy disks, after making a new cable and a simple hardware modification to the Altair disk buffer board. But you should be aware that Shugart, Mitsubishi, or other drives are much faster.

The Altair floppy disk controller outputs pulses on two lines, STEP-IN and STEP-OUT. As the names imply, a pulse on one line or the other causes the read-write head to step in or out by one track. The standard Shugart interface, however, uses lines called STEP and DIRECTION. The DIRECTION line is first set either high or low, to determine step direction, and then the STEP line is pulsed. The Tarbell, and all other soft-sector disk controllers, output these last signals.

To convert the Altair hard-sectored, 8-inch disk drives, refer to Figure 1. The circuit shown within the dashed lines is constructed via wirewrap on a small perf-board and mounted to the buffer board within the Altair drive box. The traces shown on the schematic leading to pins 10 and 12 of the Altair buffer board IC 'E' are cut. The perf-board is wired in as shown, with wire B1 going to buffer board cut-trace B1, and wire B2 going to IC 'E' pin 10, etc. Note that the second and higher drives do not have to be modified, since the Altair buffer board will transfer the now-modified signals to the drive box 'out' connector for the second and following drives.

If desired, a DPST switch may be used to make the modification switchable, allowing use of the drives with both Altair and CP/M disk controllers.

Next, a custom cable must be constructed to adapt the 50-pin connector from the controller board to the 37-pin connector on the rear of the Altair drive box. The wiring for this cable is shown in Table 2.

If using the Altair 16-MCS 16K static-memory cards, the first card (the one with address switch 1 ON) must be modified to allow it to 'phantom' when the Tarbell disk controller boot PROM is active. This modification is quite simple:

First, cut the trace from IC 'G' pin 15 to ground. I am assuming that you are holding the board with the component side toward you, and the S-100 connector down. This trace comes out from under the IC directly under the notch on the left end, and goes to the bottom leg of the filter capacitor. Cut it here. On top of IC 'E', solder a 1KΩ, ¼-watt resistor from

the top leg of the nearest filter capacitor (5 Volts) to pin 1, and another from 5 Volts to pin 2. Be careful not to overheat the IC. On the back of the board, use wirewrap wire soldered in place to connect IC 'E' pin 2 to IC 'G' pin 15. With another piece of wire, connect IC 'E' pin 1 to the S-100 connector number 67 (PHANTOM*). This completes the memory modification. The board will now properly honor the PHANTOM* signal from the disk controller. Reinstall the board.

The final hardware step is to align the Altair 8-inch drives to Shugart specifications. This is required to read the CP/M boot disk that you received with the disk controller. Head-positioning alignment should be OK, but the index-to-databurst delay must be increased from the Altair specification to about 250 ms. A bonus here is that the Shugart specification is much more tolerant, and thus the drives will not need to be realigned as often.

You will have to borrow a standard Shugart or other 8-inch drive to complete the following step, as the Tarbell CP/M will try to step the Altair drives faster than they are capable of running. (Be sure the spiral groove in the Altair stepping motor shaft is perfectly clean and very lightly lubricated with WD-40).

After drive modifications are complete, and the disk controller is installed in the Altair mainframe, connect the Tarbell-provided cable between the controller and the borrowed disk drive. Boot the system by pressing the RESET and then the RUN switches. CP/M will boot, but nothing will show on the screen, since the standard Tarbell boot disk addresses a different I/O port for the terminal. Look at the CONIN and CONOUT sections of the printed BIOS listing provided by Tarbell. Using the Altair front panel,

CP/M DRIVE	Disk Address on Altair Disk Buffer Board
Drive A	Address 14
Drive B	Address 13
Drive C	Address 11
Drive D	Address 7

Table 1. *Disk addressing of the Altair drives with CP/M.*

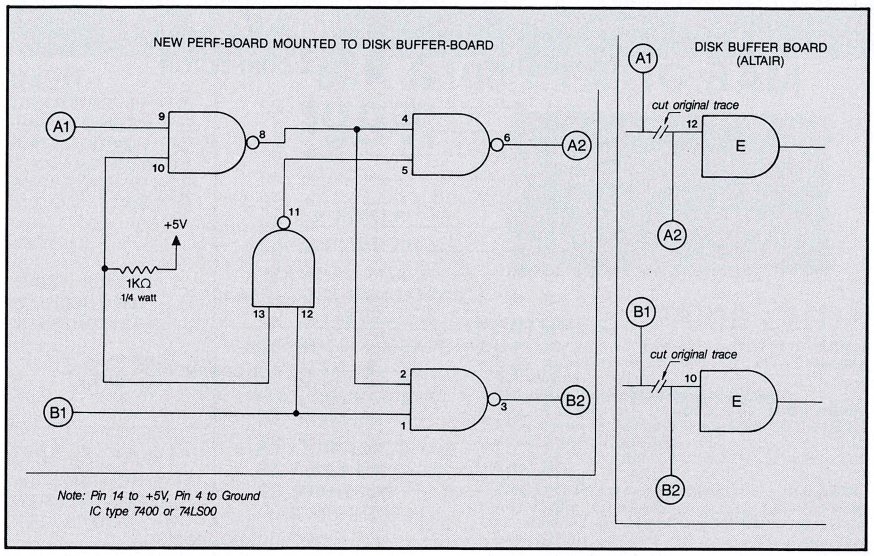


Figure 1. *Modification and extra circuit required to use the Altair's original disk drives with a Tarbell controller. See text for details.*

change the port addresses to the Altair standard (16 for status and command, 17 for data) at the proper memory locations. Change the status mask bytes also (2 for CPTR, data-ready-out; 1 for CKBR, keydata-ready). Address memory at location 0 and RUN, and you should now get the famous 'A>' prompt of CP/M.

Bring up 2DFORMAT and format a new disk. Use PIP to copy the Tarbell disk to this disk. Use WRTSYS to copy CP/M from the Tarbell disk to the new disk. Place the new disk in the A drive and press CTRL-C.

Now bring up ED, and modify the 2DBOOT.ASM and 2DBIOS.ASM files. Change the EQU (equate) called STD to FALSE. Change the TARBELL EQU to FALSE. Change the MSIO2 EQU to TRUE. Change MSIZE to 63 (if you have 64K). Change DUBSID and DMACNTL to FALSE. Change CSTAT and CCOM to 16. Change CDATA to 17. The LSTAT, LCOM, and LDATA for the Altair C-700 printer card are 2, 2, and 3 respectively. Change STEPRAT EQU to 3 (slowest, for Altair drives). Not all these EQU's are in both files. Following Tarbell's instructions, save the modified files on disk, and produce a new CP/M on the disk. Note that Tarbell will send you a disk with these modifications included at a slight extra charge, thus eliminating the need for borrowing a drive and for changing the BIOS yourself. Also note that the file names on your disk may not necessarily be the same. Tarbell also has CP/M-80 BIOS files called 512BIOS and 512BOOT. Read the instructions. If you have the 512 files, you have 512SGEN and 512FMT instead of WRTSYS and 2DFORMAT.

Replace the borrowed drive with the modified Altair drive using your custom cable, and voila, you now have an Altair CP/M system that can read and write standard 8-inch CP/M soft-sectored disks!

Tarbell formats allow 476K and 612K on single-sided disks, which by our tests have proven more reliable with the Altair drives than the standard single-density CP/M format. If interested, I can provide a listing for a program that allows using the Altair and Tarbell controllers simultaneously. This program transfers all files from an Altair disk to a CP/M disk. You must have one of each type of drive. Please send $10.00 to cover copying and mailing costs. You can also send a Tarbell double-sided or 512-density single-sided disk and $10.00, and I will place the program on the disk. It requires MBASIC under CP/M.

CONVERTING THE ALTAIR TO Z80

Another step I took with my Altair was to convert it to a 4-MHz Z80, doubling the processing speed of the Altair 8080 CPU. Besides a Z80 CPU board, this modification requires the purchase of a memory board since the Altair 16-MCS memory cards will

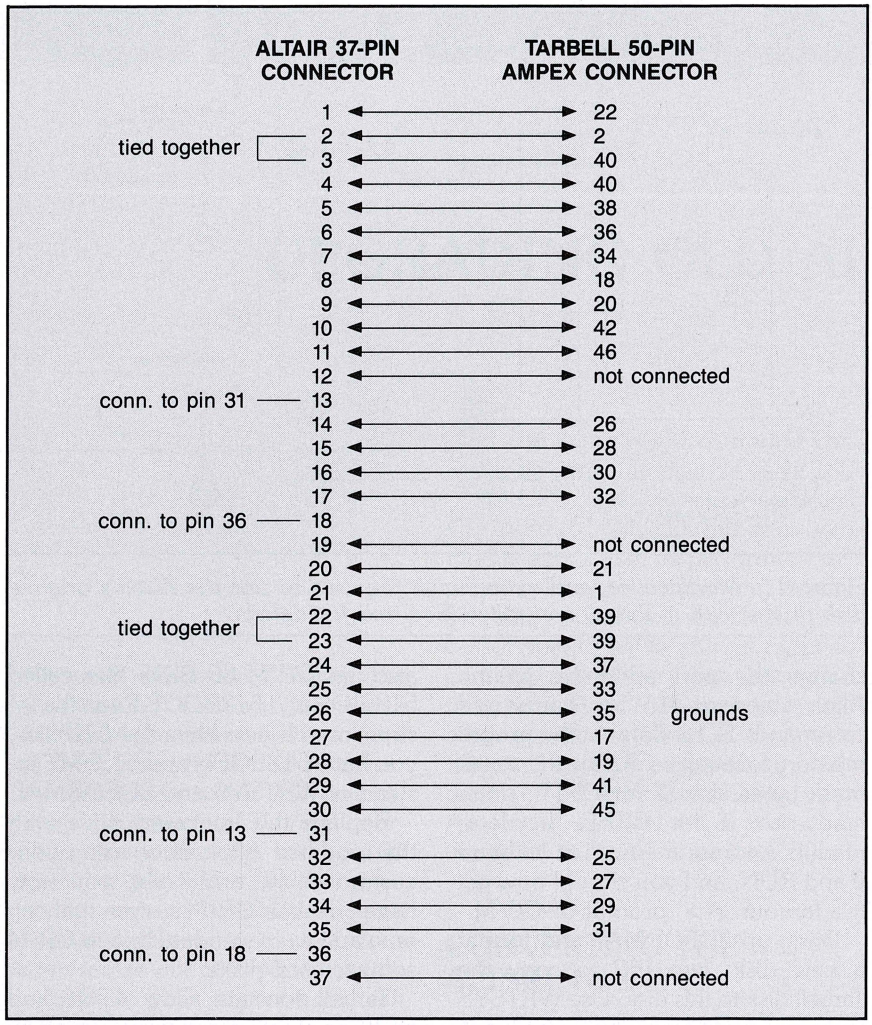


Table 2. *Cable connections from the MITS/Altair drive box to the Tarbell DD (soft sector) disk controller. Controller pins that are not listed are not connected.*

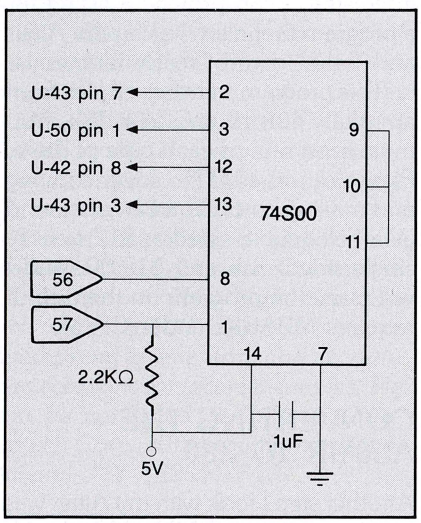


Figure 2. *Circuit required to operate the Altair front panel with the CB-2 Z80 card. Pins 4, 5, and 6 are not used. Pins 1 and 8 connect to S-100 bus lines 57 and 56 respectively.*

not work with a Z80, even though they are capable of 4-MHz operation. Any reliable S-100 64K memory card may be used as long as at least the lowest 4K block honors the standard PHANTOM* signal. I used the Fulcrum 64K card.

For the CPU, I selected Transend Corporation's CB-2 Z80 card since it has a front-panel connector and switchable speed selection. A custom cable must be made to interconnect the IC socket for the front-panel connector on the CB-2 card and the Altair panel card. Note that the schematic for the CB-2 front-panel connector socket reverses the front-panel data pin identifiers. They should be labeled 0-7 instead of 7-0. Take this into account when you make the cable.

In order to make the front panel switches functional, a single-IC circuit must be built on perf-board and mounted to the CB-2 card. Use a 74S00 IC wired as shown in Figure 2.

Also on the CB-2 board, you will find a 2-pin jumper socket between IC U-42 and the 4-position DIP switch. Leave the jumper off, and connect S-100 bus connector 58 to the upper pin. Connect the lower pin to the lower connector of switch 2 of the 4-position DIP switch. Switch 2 should be left ON and the other switches OFF.

If you will be using the Altair disk controller, also connect a wire from S-100 connector 66 (S-100 undefined line) to the upper pin of switch 3 on the 4-position DIP switch (CB-2 board). On the Altair disk controller board 1 (the board with horizontal ICs), connect a wire from S-100 connector 66 to IC 'B5' pin 8. Also connect a wire from the right side of R22 to pin 10 of IC 'B1.' These modifications will cause the CB-2 Z80 board to automatically switch from 4-MHz to 2-MHz operation during disk reads and writes only, eliminating the need for changing the software timing loops in the Altair DOS.

If you are not using CP/M, you will find that Altair BASIC does not work correctly with a Z80. Integer numbers work, but not single or double precision. Line numbers in listings always display as 00000, but GOTOs still work. It seems they used an 8080 flag improperly, which causes some very strange results with the Z80. I can provide a modified Altair Version 5-F BASIC that works on both the Z80 and 8080 for $40.00. To avoid copyright problems, you must send a disk with Altair BASIC on it, and I will return it with the modified version.

It's been a while since I made these modifications to my Altair, but I still remember the thrill of running **real CP/M programs** and at twice the speed! I hope you enjoy it too.

COMPANY ADDRESSES

FULCRUM COMPUTER PRODUCTS
459 Allan Court
Healdsburg, CA 95448

TARBELL ELECTRONICS
1082 E. Artesia Blvd., Suite C
Long Beach, CA 90805

TRANSEND CORPORATION
(formerly SSM Microcomputers)
884 Portola Rd
Portola Valley, CA 94025

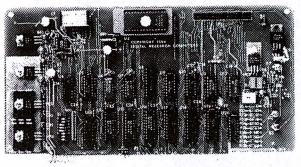
ALL SALES ARE MADE SUBJECT TO THE TERMS OF OUR 90 DAY LIMITED WARRANTY. A COPY OF THIS WARRANTY IS AVAILABLE FREE, ON REQUEST.

DIGITAL RESEARCH COMPUTERS

(214) 225-2309

S100 EPROM PROGRAMMER

OUR NEWEST DESIGN, FOR FAST EFFICIENT PROGRAMMING OF THE MOST POPULAR EPROM'S ON YOUR S100 MACHINE. COMES WITH MENU DRIVEN SOFTWARE THAT RUNS UNDER CP/M 2.2 (8 INCH). PC BOARD SET CONSISTS OF (S100) MAIN LOGIC BOARD REMOTE PROGRAMMING CARD AND SIX PERSONALITY MINI BOARDS FOR 2716, 2532, 2732, 2732A, 2764, AND 27128. SOLD AS BARE PC BOARD SET ONLY WITH FULL DOC. SOFTWARE FEATURES "FAST" PROGRAMMING ALGORITHM. FOR Z80 BASED SYSTEMS.



PC BOARD SET, FULL DOCUMENTATION, 8 IN. DISKETTE WITH SOFTWARE.

NEW! $69.95

128K S100 STATIC RAM/EPROM BOARD

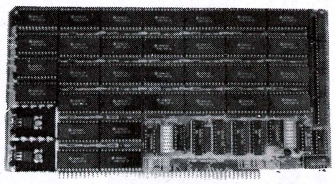
JUST OUT! USES POPULAR 8K X 8 STATIC RAMS (6264) OR 2764 EPROMS. FOR 8 OR 16 BIT DATA TRANSFERS! IEEE 696 STANDARD. LOW POWER. KITS ARE FULLY SOCKETED. FULL DOC AND SCHEMATICS INCLUDED. 24 BIT ADDRESSING.

NEW!

$59.95	$219.00	$139.00
BARE PC BOARD	128K RAM KIT	128 EPROM KIT

64K S100 STATIC RAM

$99.00 KIT



LOW POWER!

150 NS ADD $10

BLANK PC BOARD WITH DOCUMENTATION $49.95

SUPPORT ICs + CAPS $17.50

FULL SOCKET SET $14.50

FULLY SUPPORTS THE NEW IEEE 696 S100 STANDARD (AS PROPOSED)

ASSEMBLED AND TESTED ADD $50

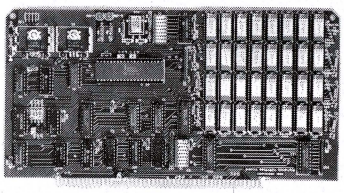
FEATURES: ***PRICE CUT!***

- ★ Uses new 2K x 8 (TMM 2016 or HM 6116) RAMs.
- ★ Fully supports IEEE 696 24 BIT Extended Addressing.
- ★ 64K draws only approximately 500 MA.
- ★ 200 NS RAMs are standard. (TOSHIBA makes TMM 2016s as fast as 100 NS. FOR YOUR HIGH SPEED APPLICATIONS.)
- ★ SUPPORTS PHANTOM (BOTH LOWER 32K AND ENTIRE BOARD).
- ★ 2716 EPROMs may be installed in any of top 48K.
- ★ Any of the top 8K (E000 H AND ABOVE) may be disabled to provide windows to eliminate any possible conflicts with your system monitor, disk controller, etc.
- ★ Perfect for small systems since BOTH RAM and EPROM may co-exist on the same board.
- ★ BOARD may be partially populated as 56K.

256K S-100 SOLID STATE DISK SIMULATOR!

WE CALL THIS BOARD THE "LIGHT-SPEED-100" BECAUSE IT OFFERS AN ASTOUNDING INCREASE IN YOUR COMPUTER'S PERFORMANCE WHEN COMPARED TO A MECHANICAL FLOPPY DISK DRIVE.

PRICE CUT!



FEATURES:

- ★ 256K on board, using +5V 64K DRAMS.
- ★ Uses new Intel 8203-1 LSI Memory Controller.
- ★ Requires only 4 Dip Switch Selectable I/O Ports.
- ★ Runs on 8080 or Z80 S100 machines.
- ★ Up to 8 LS-100 boards can be run together for 2 Meg. of On Line Solid State Disk Storage.
- ★ Provisions for Battery back-up.
- ★ Software to mate the LS-100 to your CP/M* 2.2 DOS is supplied.
- ★ The LS-100 provides an increase in speed of up to 7 to 10 times on Disk Intensive Software.
- ★ Compare our price! You could pay up to 3 times as much for similar boards.

BLANK PCB (WITH CP/M* 2.2 PATCHES AND INSTALL PROGRAM ON DISKETTE) $24.95 (8203-1 INTEL $29.95)

***CLOSE OUT!* BLANK PCB ONLY: $24.95**

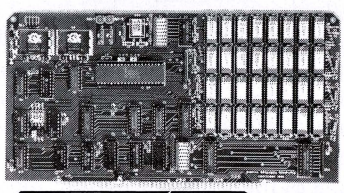
#LS-100

1 MEG. S-100 SOLID STATE DISK SIMULATOR!

WE CALL THIS BOARD THE "LIGHT-SPEED-100" BECAUSE IT OFFERS AN ASTOUNDING INCREASE IN YOUR COMPUTER'S PERFORMANCE WHEN COMPARED TO A MECHANICAL FLOPPY DISK DRIVE.

LS 100 II

NEW!



FEATURES:

- ★ 1 Meg. on board, using +5V 256K DRAMS.
- ★ Uses new Intel 8203-1 LSI Memory Controller.
- ★ Requires only 4 Dip Switch Selectable I/O Ports.
- ★ Runs on 8080 or Z80 S100 machines.
- ★ Up to 4 LS-100 boards can be run together for 4 Megs. of On Line Solid State Disk Storage.
- ★ Provisions for Battery back-up.
- ★ Software to mate the LS-100 to your CP/M* 2.2 DOS is supplied.
- ★ The LS-100 provides an increase in speed of up to 7 to 10 times on Disk Intensive Software.
- ★ Compare our price! You could pay up to 3 times as much for similar boards.

BLANK PCB (WITH CP/M* 2.2 PATCHES AND INSTALL PROGRAM ON DISKETTE) $59.95 (8203 1 INTEL $29.95)

(ADD $50 FOR A&T) **$259.00**

#LS-100 II (FULL 1 M.B. KIT)

1 MEGA BYTE!

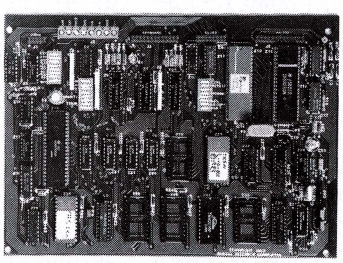
ZRT-80 CRT TERMINAL BOARD!

A LOW COST Z-80 BASED SINGLE BOARD THAT ONLY NEEDS AN ASCII KEYBOARD, POWER SUPPLY, AND VIDEO MONITOR TO MAKE A COMPLETE CRT TERMINAL. USE AS A COMPUTER CONSOLE, OR WITH A MODEM FOR USE WITH ANY OF THE PHONE-LINE COMPUTER SERVICES.

FEATURES:

- ★ Uses a Z80A and 6845 CRT Controller for powerful video capabilities.
- ★ RS232 at 16 BAUD Rates from 75 to 19,200.
- ★ 24 x 80 standard format (60 Hz).
- ★ Optional formats from 24 x 80 (50 Hz) to 64 lines x 96 characters (60 Hz).
- ★ Higher density formats require up to 3 additional 2K x 8 6116 RAMS.
- ★ Uses N.S. INS 8250 BAUD Rate Gen. and USART combo IC.
- ★ 3 Terminal Emulation Modes which are Dip Switch selectable. These include the LSI-ADM3A, the Heath H-19, and the Beehive.
- ★ Composite or Split Video.
- ★ Any polarity of video or sync.
- ★ Inverse Video Capability.
- ★ Small Size: 6.5 x 9 inches.
- ★ Upper & lower case with descenders.
- ★ 7 x 9 Character Matrix.
- ★ Requires Par. ASCII keyboard.

FOR 8 IN. SOURCE DISK (CP/M COMPATIBLE) ADD $10



$89.95 #ZRT-80 A&T ADD $50

(COMPLETE KIT, 2K VIDEO RAM)

BLANK PCB WITH 2716 CHAR. ROM. 2732 MON. ROM $49.95

SOURCE DISKETTE - ADD $10

SET OF 2 CRYSTALS - ADD $7.50

THE NEW 65/9028 VT ANSI VIDEO TERMINAL BOARD!

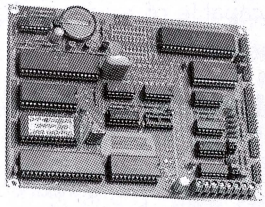
★ FROM LINGER ENTERPRISES ★

A second generation, low cost, high performance, mini sized, single board for making your own RS232 Video Terminal. This highly versatile board can be used as a stand alone video terminal, or without a keyboard, as a video console. VT100, VT52 Compatible.

FEATURES:

MICRO SIZE!

- ★ Uses the new CRT9128 Video Controller driven by a 6502A CPU
- ★ On-Screen Non-Volatile Configuration
- ★ 10 Terminal Modes: ANSI, H19, ADM-5, WYSE 50, TVI-920, KT-7, HAZ-1500, ADDS 60, QUME-101, and Datapoint 8200
- ★ Supports IBM PC/XT, and Parallel ASCII Keyboards
- ★ Supports standard 15.75 kHz (Horiz.)
- ★ Composite or Split Video (50/60 Hz)
- ★ 25 X 80 Format with Non-Scrolling User Row
- ★ Jump or Smooth Scroll
- ★ RS-232 at 16 Baud Rates from 50 to 19,200
- ★ On Board Printer Port
- ★ Wide and Thin Line Graphics
- ★ Normal and Reverse Screen Attributes
- ★ Cumulative Character Attributes: De-Inten, Reverse, Underline and Blank
- ★ 10 Programmable Function Keys and Answerback message
- ★ 5 X 8 Character Matrix or 7 X 9 for IBM Monitors
- ★ Mini Size: 6.5 X 5 inches
- ★ Low Power: 5VDC @ .7A, ± 12VDC @ 20mA.



$99.95 FULL KIT

w/100 Page Manual

ADD $40 FOR A&T

OPTIONAL EPROM FOR PC/XT STYLE SERIAL KEYBOARD: $15

SOURCE DISKETTE: PC/XT FORMAT 5¼ IN. $15

Digital Research Computers

P.O. BOX 381450 • DUNCANVILLE, TX 75138 • (214) 225-2309

TERMS: Add $3.00 postage. Orders under $15 add 75¢ handling. No C.O.D. We accept Visa and MasterCard. Tex. Res. add 6-1/4% Tax. Foreign orders (except Canada) add 20% P & H. Orders over $50 add 85¢ for insurance.

*TM OF DIGITAL RESEARCH INC. (CALIF.)

WE ARE NOT ASSOCIATED WITH DIGITAL RESEARCH INC. (CALIF.) THE SUPPLIERS OF CPM SOFTWARE

A LITTLE BARCODE MAGIC FOR THE S-100 SYSTEM

Burt Hanagami

Without a doubt, barcodes have flourished and have become a very important part of our lives. We see them almost everywhere, in supermarkets, in libraries, factories, and even in our homes. Barcodes are very inexpensive to print, and most of the required hardware is readily available off-the-shelf. If you own a S-100 with a good dot-matrix printer and a spare parallel port, you're almost there. All you need is a scanning wand, suitable cabling, and some creative software. In this article, I will show how to interface a scanning wand to your S-100 CP/M system. After installing the wand and the driver software described here, any application program that reads from the keyboard will be capable of reading a barcode.

HP BARCODE WAND AND DOT-MATRIX PRINTER

The two most critical components of the barcode system are the barcode wand and the dot-matrix printer. I have selected the Hewlett-Packard HEDS-3000 barcode wand because it is relatively easy to interface. It only has three wires: 5 volts, ground, and a TTL digital output.

The HP wand has a button that must be pressed when reading. This is a very handy feature because all the noise is eliminated when the button is not being pressed. The wand is expensive (about $115.00) but worth every penny. My only critism is that the plastic tip has a tendency to wear from repeated contact with the paper. I found that a piece of scotch tape on the worn spot, changed on a regular basis, seems to alleviate this problem. Hewlett-Packard is aware of the problem and is willing to provide additional tips at a nominal charge. Replacement is easy because the plastic tip screws on and off like a bottle cap.

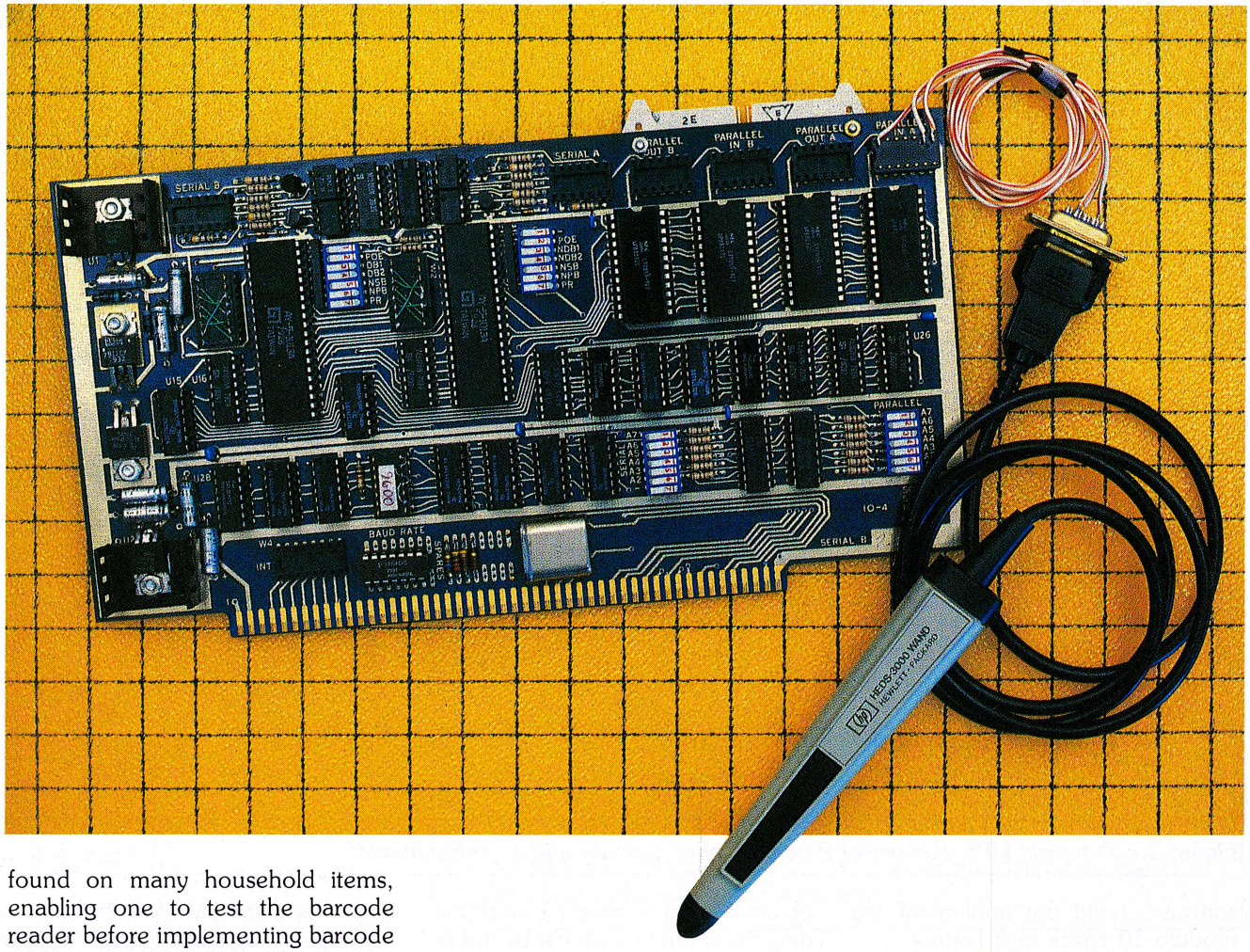
The important thing to consider in a dot-matrix printer, outside of graphic capabilities (the ability to print each dot independently), is its registration (or repeatability) and the spacing between the dots. Registration is important because multiple passes will have to be made to obtain a clean, crisp image. Paper on a sprocket-fed printer tends to have more horizontal shifting than paper on a printer with a friction platen. Also, many printers are bidirectional (the print head prints forwards and backwards). Quite often, a particular dot printed with the head moving in one direction will be slightly off from the same dot printed with the head moving in the other direction. A little slop is actually desirable, as long as the edges of the bars are smooth, since it produces an overlapping dot effect. However, excessive slop causes the dots to be fatter, or even to appear double, diminishing the spacing between different dots. Black bars in general tend to appear to the barcode wand slightly wider than white bars, although they are exactly the same width. This is due to the hysteresis effect and the filtering electronics in the barcode wand.

I have used the IDS 460G printer (with a 9-wire print head) with excellent results.

THE UPC FORMAT

I decided to standardize the barcode on the UPC (Universal Product Code) format because it is the simplest to understand and the easiest to implement. The barcode labels are ►

Burt Hanagami is a Member of the Technical Staff at Rockwell International, specializing in interrupt-driven and multitasking software. Burt also likes to invest in real estate and to work on automobiles.



found on many household items, enabling one to test the barcode reader before implementing barcode printing. To make my barcode system as versatile as possible, I incorporated the software into the operating system, so that the barcode wand would appear to be hooked up in parallel with the keyboard. This way, one can write software in practically any language without having to write custom drivers (the wand even works fine with Ashton-Tate's dBase II). There is only one constraint: the inputs from the wand must be twelve digits (a condition imposed by the UPC format) followed by a carriage return. The software is capable of scanning the barcode forwards and backwards.

The UPC (Universal Product Code) format consists of a set of 3 narrow guard bars (2 black, 1 white), six digits, a set of 5 guard bars, six digits, and another set of 3 guard bars. Figure 1 shows a typical barcode. The guard bars help define the start and the end of the barcode label, and can help compensate for varying scanning speeds. All the guard bars, white and black, are of identical width. This width is the basic unit of the barcode. All the other bars are multiples of this width. Note that the set of guard bars in the middle has an odd number of bars. This enables the left guard bars to start with a black bar, and the right guard bars to end with a black bar. The bars representing the six digits on the left side are reversed in color (black is white and white is black) when compared to the six digits on the right side.

All UPC barcode labels have exactly 59 black and white bars, and they always start and end with a black bar. On the typical barcode label, the guard bars extend slightly lower than most of the other bars. Each digit is represented by two white and two black bars. The width of a digit is seven guard bar units. The digits are also displayed in character form. There is usually a single digit to the very left of the label (commonly a 0 or a 3), five digits on the left side, under the bars, and five digits on the right side, under the bars. The very first digit was intended for expansion purposes and was originally always set to 0. Today, sometimes labels can be found with a 3, or very rarely, a 4. For example, the Band-Aid brand of bandages from Johnson & Johnson has a 3. The next five digits are the manufacturer's code. The five digits after the middle guard bars represent the generic product code.

The last unannotated number is the modulo 10 check digit. For error checking purposes, the last number was specially selected to be equal to the remainder of the first eleven digits added together and divided by 10. If the first eleven digits were 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, and 1, their sum would be 46. Then 46 divided by 10 would be 4 with the remainder 6. Thus the check digit would be 6, the remainder. If any of the first eleven digits were misread, the total would be something other than 46, and the modulo 10 check digit would not match. In my barcode-decoding

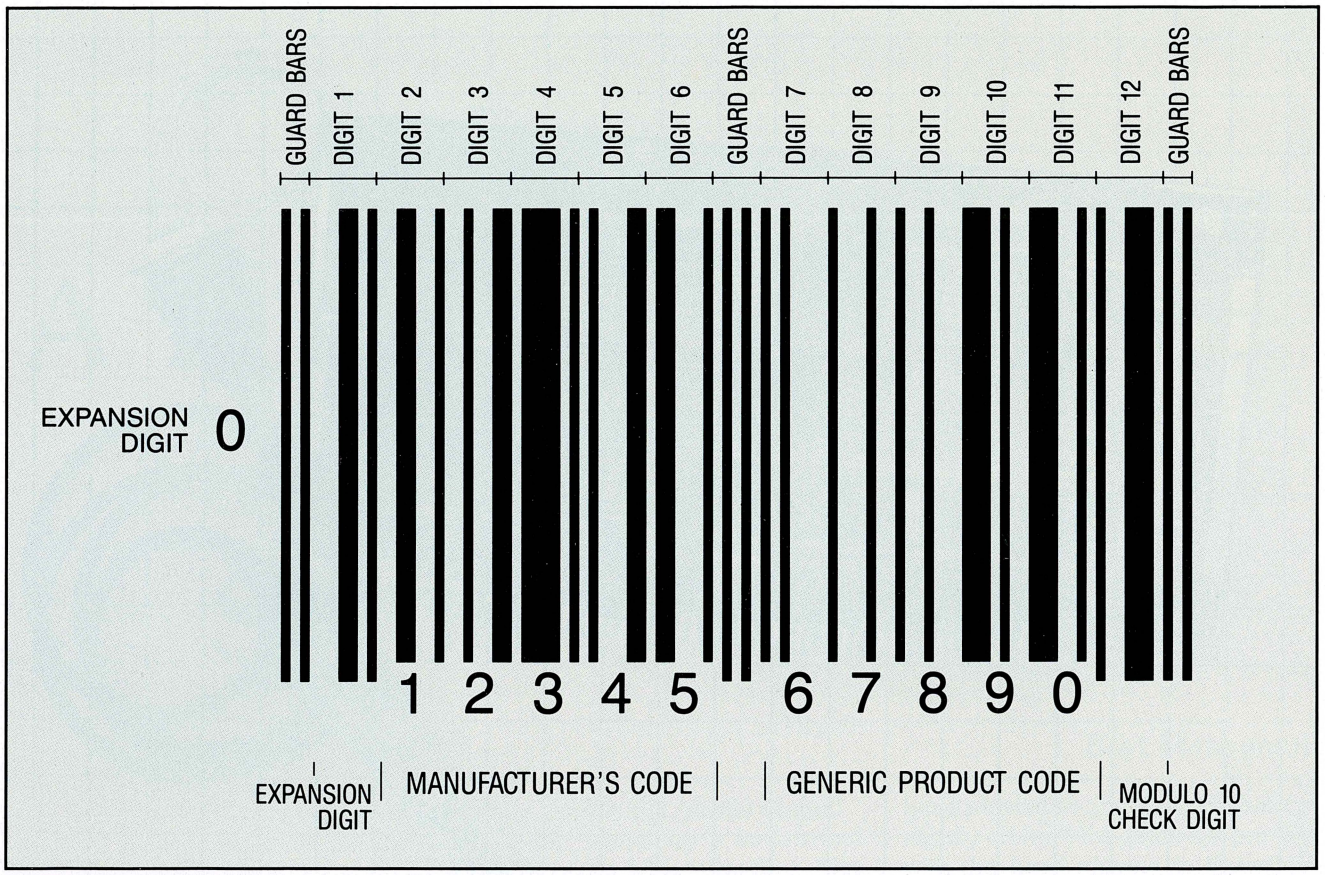


Figure 1. *A typical UPC (Universal Product Code) barcode and its components.*

software, I did not implement the modulo 10 check digit feature.

DECODING THE BARCODE

The following steps describe how I decided to decode the bars.

Step 1. Read in the barcode, recording the width (in counts per bar) of all 59 bars in a bar-width table. For each digit, the color of the bar is not important, only the width. As described in the UPC format, Digit 1 is represented by bars 4 through 7, Digit 2 by bars 8 through 11, and so on. Before we arrive at digit 7, we have to skip over the five middle guard bars, so digit 7 would occupy bars 33 through 36. Because of the reversed nature of the left and right sides of the barcode, when we compare a digit value 0 on the left side against a digit of value 0 on the right side, we see the relationship depicted on Table 1.

Step 2. Add the (time-measured) widths of the four bars that make up each digit, and divide by seven. Since a digit is seven units wide, the result obtained must represent 1 unit. Dividing the width of each bar by this result gives the width of the bar in standard units. This technique provides a dynamic way of compensating for scanning speed variations on the fly.

Step 3. Transform the data in the bar-width table into standard units. A bar may be 1, 2, 3 or 4 units in length. The transformed data is stored in a standard-unit table.

Step 4. The standard-unit table is examined, and the digits are decoded from the left-to-right and the right-to-left directions. Table 2 shows how the barcode digits are decoded. A value 0 is the bar-width sequence of 3-2-1-1, a value 1 is 2-2-2-1, etc. The label may be scanned from either direction. However, note that if a

BAR	LEFT SIDE OF BARCODE	RIGHT SIDE OF BARCODE	LENGTHS
FIRST	3-UNIT-WIDE WHITE BAR	3-UNIT-WIDE BLACK BAR	3
SECOND	2-UNIT-WIDE BLACK BAR	2-UNIT-WIDE WHITE BAR	2
THIRD	1-UNIT-WIDE WHITE BAR	1-UNIT-WIDE BLACK BAR	1
FOURTH	1-UNIT-WIDE BLACK BAR	1-UNIT-WIDE WHITE BAR	1

Table 1. *Bar lengths and colors for a digit of value 0.*

left-to-right scan yields the bar sequence 3-2-1-1, then a right-to-left scan of that bar digit would yield the sequence 1-1-2-3 which is not a valid sequence. The codes in the UPC barcode are specially selected so that reversing the bar sequence of a valid number cannot be interpreted as another valid number.

To speed things up, the bar sequences are converted into 8-bit hexadecimal numbers. A bar length of 1 would be 01, a 2 would be 10, a 3 would be 11, and a 4 would be 00. A 3-2-1-1 sequence is first reversed, then converted to '01 01 10 11' or 05BH (hexadecimal notation).

If a decoded value matches the value of one of the ten digits, that digit is recorded in the answer tables. There are two answer tables, one for each direction. If no match is found, a '?' is inserted.

Step 5. If a valid scan is made, one of the answer tables should contain all the digits in the barcode, and the other answer table should be all '?.' If a table contains some digits and some '?,' the scan was not successful.

DIGIT	LEFT-SIDE BARCODE	RIGHT-SIDE BARCODE	SEQUENCE	HEX (See Text)
	1 2 3 4 5 6 7	1 2 3 4 5 6 7		
0			3 2 1 1	5B
1			2 2 2 1	6A
2			2 1 2 2	A6
3			1 4 1 1	5D
4			1 1 3 2	B5
5			1 2 3 1	79
6			1 1 1 4	D5
7			1 3 1 2	9D
8			1 2 1 3	D9
9			3 1 1 2	97

Table 2. *Method used to decode the digits of a barcode.*

HARDWARE COMPONENTS

1. S-100 system, complete with disk drives, parallel port, CP/M-2.2, an 8080 assembler, and BASIC (for printing barcodes).
2. Dot-matrix printer to print barcodes.
3. HEDS-3000 wand from Hewlett-Packard.
   (Radio Shack Computer Centers also carry a wand that appears to be identical to this one. Part No. CA 26-1183A. Price $99.95.)
4. Cable from the parallel port to the wand. The only piece of hardware that will have to be built.

Table 3. *Components required to read and print barcodes.*

THE HARDWARE

The hardware was the easiest part of this project. The necessary components are described on Table 3.

For the parallel interface, I used a SSM (now Transend Corp.) IO-4 board with the parallel port addressed at 0A4H. This address is set with the switches as follows:

A7, A5, and A2 — OFF
A6, A4, A3, and A1 — ON

A 3-wire adaptor cable connects Parallel Input A (14-pin header socket on the SSM IO-4) to the DB-9 connector in the wand. Refer to Table 4 for the cable wiring.

If you plan to use a different board, you will have to select a parallel port bit that can be initialized as an input, and connect it to the TTL IN signal (pin 2 of the wand DB-9 connector). The wand is TTL compatible, so buffering should not be necessary. The power and ground connections (pins 9 and 7) should be the same power and ground driving the parallel port chip.

THE SOFTWARE

The software was developed in four separate programs, WAND, CWAND, PATCH, and BARCODE.BAS. The program listings start on page 28. WAND is a test driver that I used to fine tune the software algorithms. Any modifications are transferred to CWAND (CP/M Wand). CWAND is the actual software driver. PATCH is used to install CWAND. And BARCODE.BAS is a BASIC program used to generate the barcode labels.

SIGNAL	Parallel Port on SSM IO-4	DB-9 Connector on Wand
+5V	PIN 14	PIN 9
GND	PIN 8	PIN 7
TTL IN	PIN 9	PIN 2

Table 4. *How to connect the cable from the wand to the parallel port.*

WAND, the Test Program

The WAND program reads the barcode label in the manner described above and displays a list of timing data as shown in Table 5. The first, the eighth, and the fifteenth lines of data are for the guard bars. The wand accelerated from 32H (50 decimal) to 0BH (12 decimal) counts per guard bar units during this scan. On the other (digit) lines, the first four numbers are the counts for the four bars. The next four numbers are the data (representing each bar width) reduced to guard bar units, and the last four-digit number is the special hex number used to decipher the digit. The left-to-right order in the bar-width data is the least significant to the most significant (backwards). There-

```
# OF BARS = 003B
0032  0027  0023
0025  0045  0050  0028    0001  0002  0003  0001    0079  DIGIT=5
001C  006E  0021  001A    0001  0003  0001  0001    005D  DIGIT=3
001A  002D  0046  001E    0001  0002  0003  0001    0079  DIGIT=5
003A  002D  0016  0012    0003  0002  0001  0001    005B  DIGIT=0
0035  0028  0017  0013    0003  0002  0001  0001    005B  DIGIT=0
0032  0027  0014  0010    0003  0002  0001  0001    005B  DIGIT=0
0010  0010  0010  000E 0010
001E  001D  0023  0010    0002  0002  0002  0001    006A  DIGIT=1
000D  000F  002A  001D    0001  0001  0003  0002    00B5  DIGIT=4
002E  001E  0012  000F    0003  0002  0001  0001    005B  DIGIT=0
000E  001B  002D  0012    0001  0002  0003  0001    0079  DIGIT=5
000C  000E  000D  0034    0001  0001  0001  0003    00D5  DIGIT=6
0010  0026  0010  0017    0001  0003  0001  0002    009D  DIGIT=7
0010  000C  000B
```

Table 5. *Example output of the WAND program. See text for explanation.*

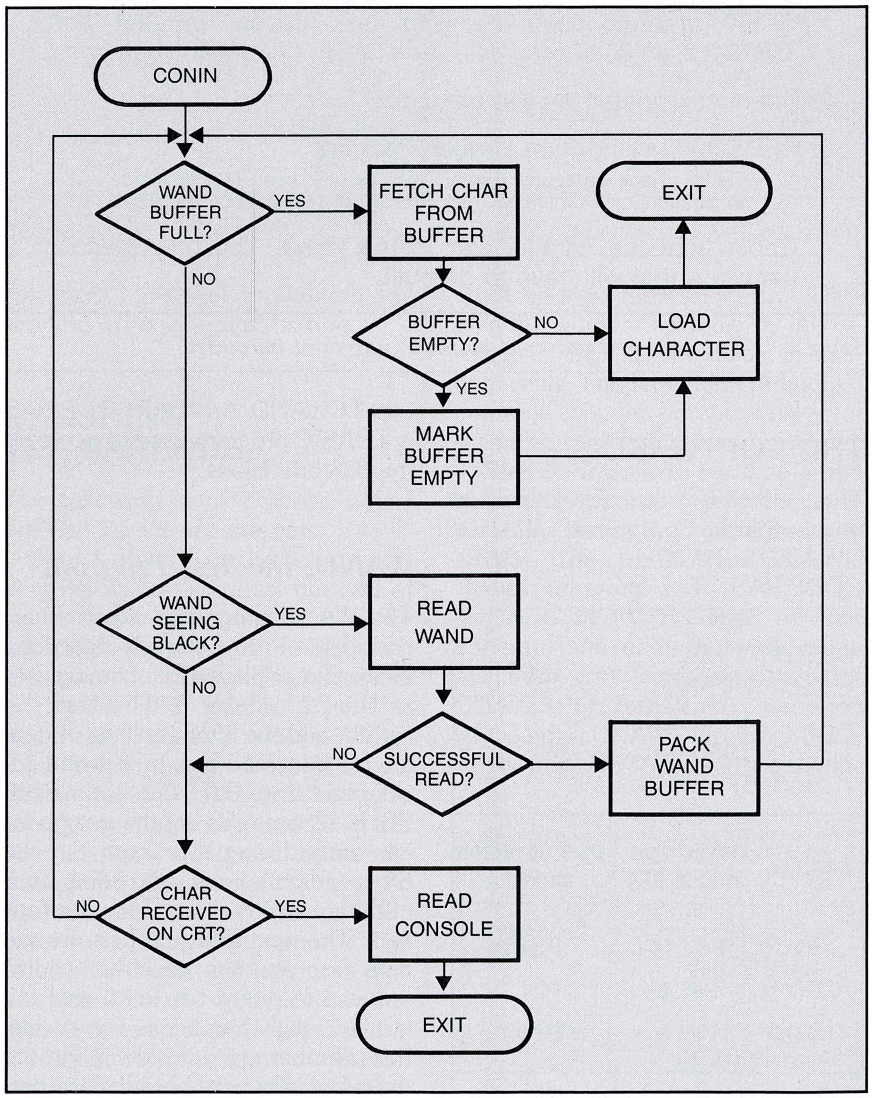


Figure 2. *Flowchart of CWAND, the driver program for the barcode wand.*

fore, the sequence '0001 0002 0003 0001' would convert to '01 11 10 01,' or 79H. Notice that 0003 is third in the sequence while its binary counterpart, 11, is second in the sequence.

This program should be used to help integrate, troubleshoot, and fine tune the software to match your hardware. Different S-100 systems run at different speeds, barcode labels printed on different printers vary, and everyone has a different scanning technique. Any changes made to the WAND program should be ported over to the CWAND program.

If your S-100 system does not have a SSM IO-4 board, you may have to:

1. Add parallel port initialization (if not done by the operating system) in the initialization section of WAND.

2. Examine the BSTRIP (black strip) and WSTRIP (white strip) routines. For most parallel devices, you would only have to modify the WAND, BLACK, and WHITE equates. WAND is the data port number, BLACK is the value read on the parallel port when the wand is over a black bar, and WHITE is the value read when the wand is over a white bar. The difference between BLACK and WHITE would be the state of the bit on the parallel port connected to the wand. White on the Hewlett-Packard HEDS-3000 wand is low or a 0, and black is high or a 1.

CWAND, the Actual Driver

The CWAND program is designed to overlay the CONIN (console input) driver in the CP/M BIOS (Basic I/O System). The basic driver flowchart is shown in Figure 2. The logic is a bit tricky. The main loop checks if the barcode buffer is full, if the wand is scanning, or if a key has been pressed.

When a key is pressed, the driver operates normally, and it reads the keyboard. When the wand is scanning (indicated by a black bar input), the driver reads the barcode in much the same manner as the WAND program. If the driver reads the barcode successfully, it receives twelve numbers. During the actual barcode scan, the keyboard is not scanned. Special timeout loops were incorporated to prevent lock up. The original driver handles one character at a time, so the numbers are stored in a buffer. A

carriage return is packed at the tail end of the buffer to terminate the sequence. The driver will unpack one character each time it runs, until the buffer is empty. A transient program calling the BIOS driver thinks that it is only reading the keyboard. It does not know about the barcode wand.

The CWAND program must be customized to your system as follows:

1. Transfer the changes made in the WAND program over to this program.
2. If you had to add a parallel port initialization routine in WAND, find another place to put it. The ideal location is where it would only be executed once.
3. The proper way would be to integrate this driver into the BIOS section, reassemble, and generate a new CP/M operating system. The CWAND driver replaces the CONIN driver and the CONST driver. If you generate a new CP/M system, you may ignore the PATCH program.

PATCH, the Installation Program

The PATCH program was designed for S-100 systems that do not have enough room in their boot tracks to accommodate the CWAND driver. The PATCH program copies the CWAND driver into free memory, and modifies the warm boot (location 0001H and 0002H) jump address to where the CWAND driver actually resides. The first instruction in CWAND contains a jump to the real warm boot location. The CWAND driver is loaded by the PATCH program just below CCP (Console Command Processor) in CP/M.

If you do not have enough room in your system's boot tracks, or simply do not wish to regenerate your operating system, you will have to perform the following procedures.

1. Reassemble the CWAND program to load just below the CCP.
2. Modify the PATCH program to reflect the proper addresses.
3. Using DDT (Dynamic Debugging Tool): a) Load the PATCH program at location 0100H. b) Load the CWAND program at location 0200H. c) Save object code in a file called PATCH.COM.

These procedures require an intimate knowledge of the CP/M operating system, as well as the specifics of the CP/M implementation on the S-100 system in question. If you have doubts on how to proceed, you may want to consult with someone who knows CP/M well.

BARCODE.BAS Prints the Barcode Labels

The BARCODE.BAS is a BASIC program that prints the barcode. Subroutines located at statements 500, 600, and 700 must be adapted to your specific printer. These subroutines are *'turn graphics on', 'turn graphics off,'* and *'print the graphic frame,'* respectively. A few lines past statement 200 is an ASCII description that is printed with the label in condensed print. This too will have to be adapted. You will need to consult your printer manual for the details.

SCANNING WITH THE WAND

To scan with the barcode wand, hold it like a pencil, with index finger resting over the wand button. Position the wand on the paper, 1 or 2 inches to the left (or right) of the barcode label. Press the wand button, and scan the label. Positioning the wand a couple of inches away allows the wand to be up to full acceleration by the time the barcode label is reached.

The barcode drivers will always be in residence if you installed the barcode system in BIOS. But, if you went the patch route, you have to run the PATCH program before you run an application program. In either case, your application program only needs to know how to read from the console; it will automatically access CWAND when you start scanning.

COMPANY ADDRESSES

HEWLETT-PACKARD
1000 N. E. Circle Blvd.
Corvallis, Oregon 97330

TRANSEND CORPORATION
(formerly SSM Microcomputers)
884 Portola Rd.
Portola Valley, CA 94025

S-100 BOARDS

■ **DUPLEX 816:** 8 MHz 80186 & Z80H CPUs, 2 RS232 Serial Ports, 5 & 8-inch Floppy Interface, 8 Kbyte ROM, Memory Management to 16 Mbytes & Interrupt Handler. Runs PC-DOS, CP/M-80, CP/M-86, and Concurrent CP/M-86. CP/M-80 vs 2.2 included. Price: $995.

■ **CPU/IO:** 4 or 6 MHz Z80, 2 RS232 Serial Ports, Memory Management to 1 Mbyte, Programmable Timer, Vectored Interrupt. Price: $316.

■ **DD-ASM:** Double Density Floppy Disk Interface, including boot ROM, Direct Memory Access, CP/M 2.2 operating system. Price: $316.

■ **FDI-ASM:** Single Density Floppy Disk Interface, including boot ROM. Assembled: $260, Kit: $180.

■ **4S/2P:** Four RS-232 Serial and two Parallel ports. Price: $360.

SOFTWARE

■ **DB-3:** The Tarbell Database System, for PC-DOS, CP/M-80, CP/M-86, or Concurrent CP/M-86, Multi-User File Locking, Interactive Program Generator, Up to 19 Files Open at Once, Extensive HELP system, Multi-Field Sort, and Multi-Line Field Type, Price: $95.

■ **CONCURRENT DOS:** Multi-User, Multi-Tasking Operating System for Duplex 816 Above. Has Virtual Terminals, PC-MODE, and Many Utilities, Price: $600.

■ **CP/M-80 vs 2.2:** Industry Standard Operating System for 8080, 8085, or Z80. Price: $75.

■ **INV:** Manufacturing Inventory System, written in dBASE-II. Items may consist of as many levels of other items as desired. Menu includes buying, receiving, packaging, shipping, listing, totaling, and others. Price of source: $80.

■ **SALES:** Sales Order Entry System, written in dBASE-II. Menu includes order entry, change order, list orders, print invoice, and others. Price of source: $80.

■ **TBAS:** Tarbell Disk BASIC Interpreter. Has many extra functions, such as high-speed file search, I/O reassignment, procedures, and alphanumeric labels. Price: $49.

■ **PCPROM-SRC:** Assembly-Language source for our own ROM-BIOS that's used on the Duplex 816 to boot up PC-DOS directly. Price: $150.

TARBELL ELECTRONICS
1082 E. Artesia Blvd., Suite C
Long Beach, CA 90805, (213) 422-7081

BARCODE MAGIC FOR S-100

LISTING 1

WAND — THIS IS A TEST PROGRAM TO DRIVE THE BARCODE WAND.

If you are not using the SSM IO-4 board, add the parallel port initialization code if needed. Also, examine the BSTRIP and WSTRIP routines; for most parallel devices, only the WAND, WHITE, and BLACK equates need modification.

```
0100            ORG     0100H   ;TPA
0005 =          BDOS    EQU     0005H           ;BDOS
0001 =          CONINP  EQU     0001H           ;CONSOLE INPUT
0002 =          CONOUT  EQU     0002H           ;CONSOLE OUTPUT
0009 =          PRINTF  EQU     0009H           ;PRINT MESSAGE
        ;;;
```

S-100 BASED LOCAL AREA NETWORKS

A HARDWARE PRIMER

Jay Vilhena

John Martin is an accountant. A few years ago, when he went into business for himself, John knew that one of his first investments had to be computer equipment. Knowing nothing about computers, he decided to play it safe by buying a brand-name IBM PC. Now John has two secretaries and one assistant. He has since assigned the PC to one of his secretaries, bought an IBM AT for his own office room, and bought two clones for the other two employees.

John's list of clients has also grown. He has three 10-Meg hard disks and a Bernoulli box to store all his software and clients' data. Often, John has to work on the same data as his assistant or his secretaries. Sometimes they make two working copies of a database, but that is usually inconvenient because it is easy to forget which copy is the most recent. Most often, they simply carry a floppy around the office to the next person who needs to work on it, or to print a letter on the daisy-wheel that sits next to John's AT.

Several years ago, Blueridge High School acquired a 16-user system to teach a small class in computer applications. The computer was installed by a now-defunct, local systems house that purchased S-100 boards and integrated them into multiuser systems according to clients' requirements. Lately the school administration, following industry trends, has been buying IBM PCs and compatibles, and a few Macintoshes. Several of these machines populate not only classrooms but also several offices throughout the building. Some have laser printers attached, others hard disks, and one a color plotter.

The school's old S-100 multiuser still gets its share of classroom use. It has always operated reliably. However, the school feels that the students should instead be exposed to newer-generation software which does not run on the old S-100. The administration has been considering junking the machine and buying more PCs and Macs. Still, that means buying more printers, disk drives, etc.

In both John Martin's office and Blueridge High School, two real-life examples, the vital need to network is obvious. In addition, the school does not realize that S-100 architecture is upgradable and that the old multiuser can not only be made to run newer software but also provide the means to network all the other systems and save bundles of taxpayers money on peripheral equipment.

A network is, number one, a means of sharing information, number two, a means of sharing peripherals, and number three, a flexible way to assign/deassign computer resources to individuals in a group of users. With a local area network (LAN), everyone in John's office can access the same copy of a database. John needs to purchase a good S-100 computer and hook up a large hard disk and his fastest or unique printers to it. The S-100 will provide the modular means for future upgrading. The PCs can all be connected to the S-100 (the network server) and all the users will be able to access files and peripherals on the server. The single copy of the database only needs to reside on the server's hard disk. Furthermore, John can decide which files and programs each person should or should not be able to access.

Similarly, Blueridge High School can tie all its computers together, hundreds of them if necessary, and ►

Jay Vilhena is Editor of S-100 Journal. In his spare time, which lately appears to be evermore elusive, he engages in painting, skiing, and marine biology.

A. S-100 LAN CARD

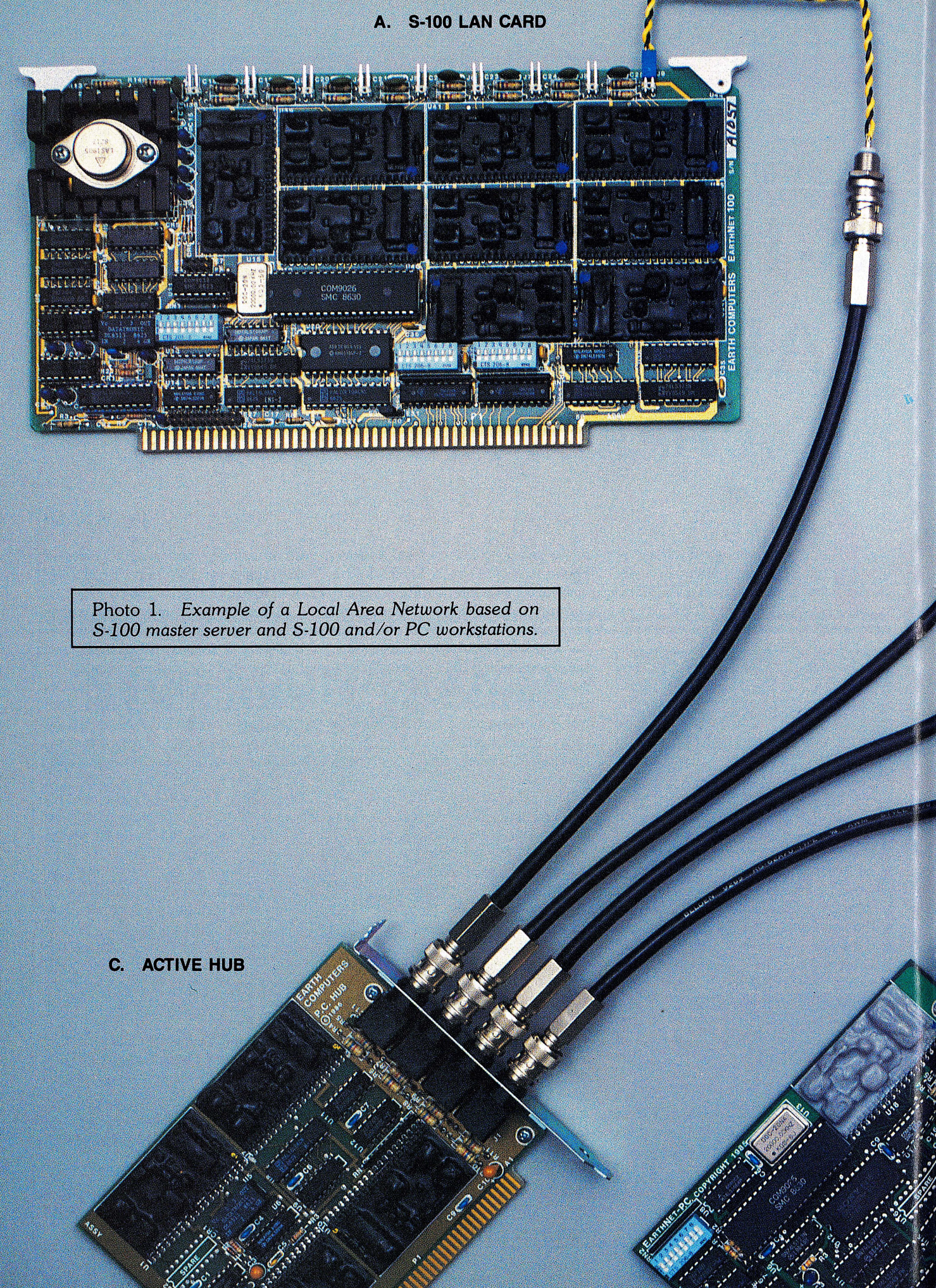
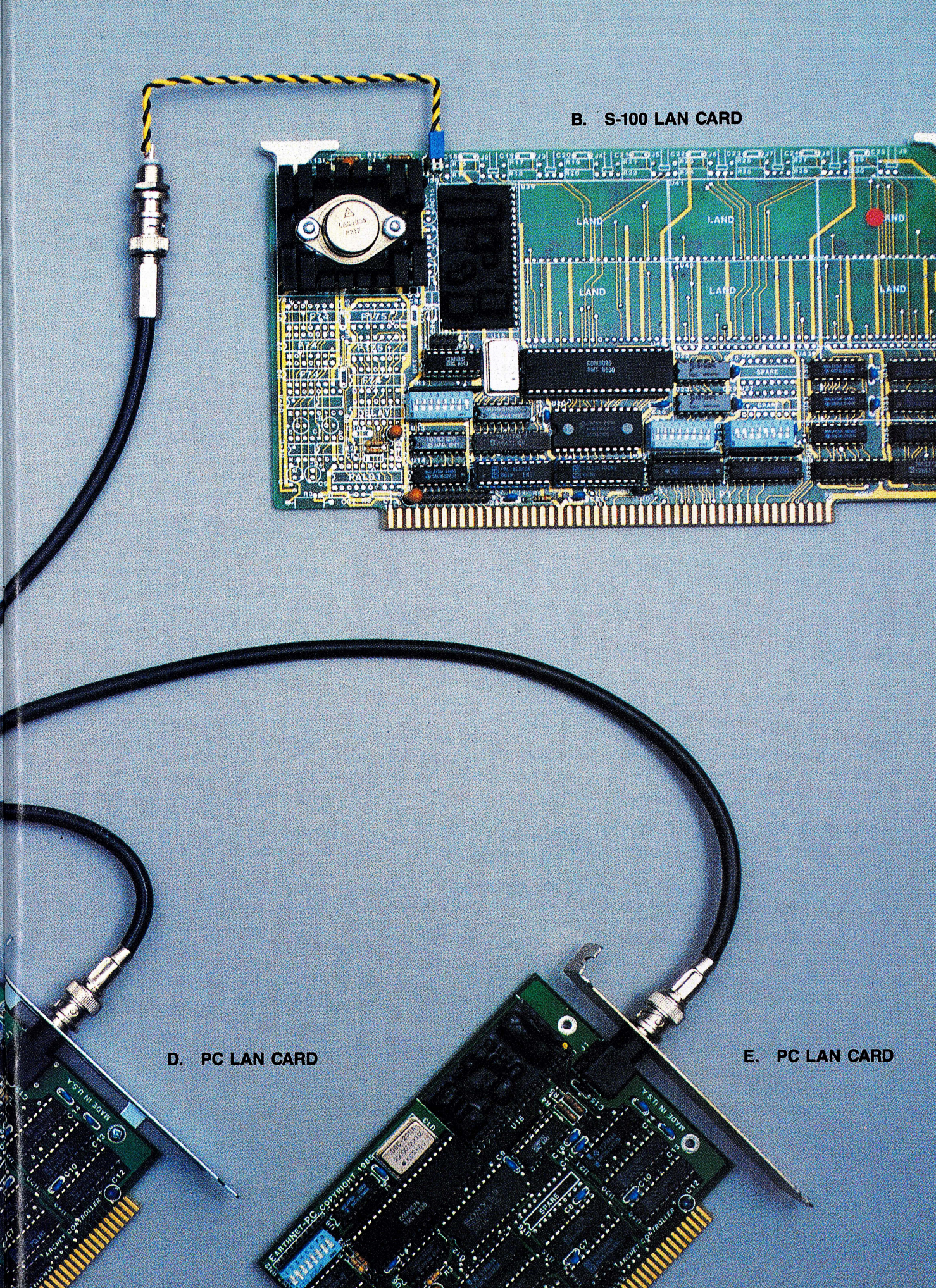


Photo 1. *Example of a Local Area Network based on S-100 master server and S-100 and/or PC workstations.*

C. ACTIVE HUB


B. S-100 LAN CARD
D. PC LAN CARD
E. PC LAN CARD

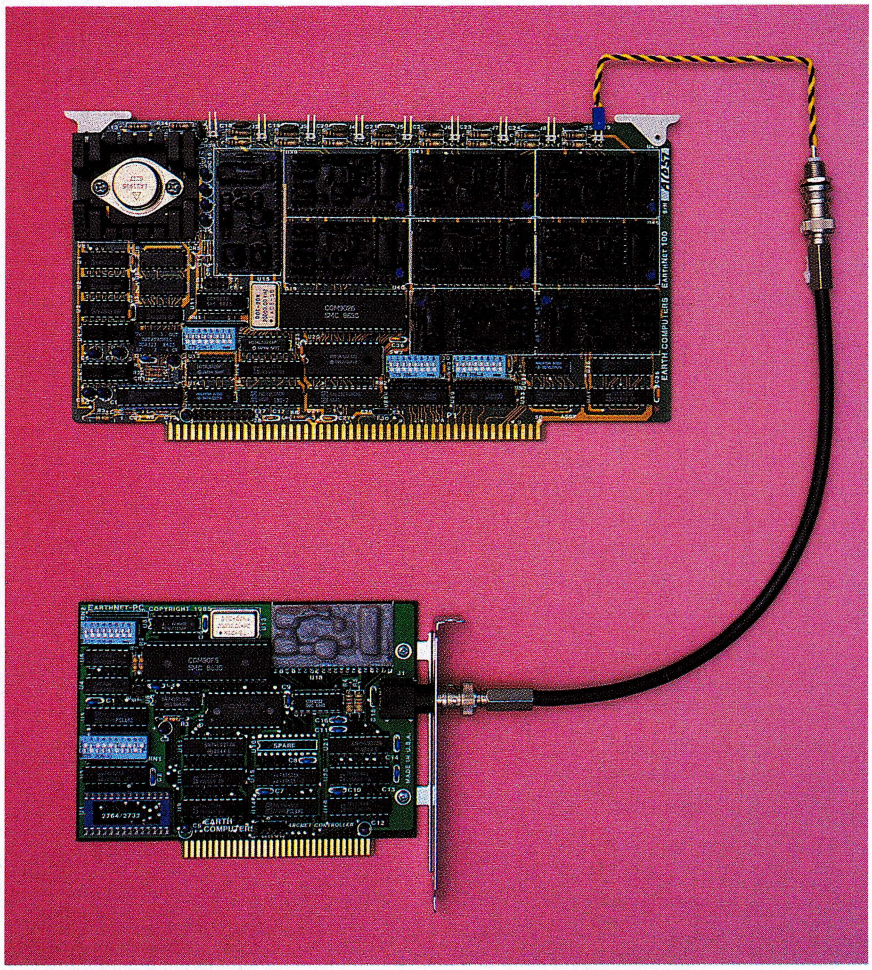


Photo 2. *The simplest LAN is a connection between two computers. The hardware shown would allow connecting a PC as a workstation to an S-100.*

they will all be able to access laser printers placed in strategic locations, the plotter, any other peripherals, and one (or more) central, large-storage hard disk connected to the main S-100 LAN server.

Many other businesses, departments, and individuals need networks. In many cases, the underlying hardware already exists, needing only a few upgrades and some software. Recognizing the need for networks and the high suitability of S-100 systems as network servers or controllers, S-100 Journal publishes articles that address questions related to implementing and using S-100-based LANs. In an article by Akin Orhun in S-100 Journal No. 4, we presented 14 pages of information discussing many of the concepts and details associated with LANs. In this article, I will explain the hardware details of what is involved in tying S-100s and PCs together. Other readers who feel they might be able to write a good article related to S-100s and LANs are invited to contact me at S-100 Journal (phone and address on page 1).

A NETWORK INSIDE A BOX

Before discussing the S-100/PC network, and so that you are aware of the various alternatives, let's take a look at the classical S-100 network. Long before the term *network* became the vogue, the S-100 bus internal architecture was designed in a way that amounts to an internal network.

The S-100 bus allows multiple CPU boards. These various CPUs can be similar or totally dissimilar. For example, Motorola 68000s, Zilog Z80s, or Intel 80286s can all be present inside an S-100 frame. They all communicate with each other through the S-100 motherboard (the backplane that the S-100 cards plug into). They can share memory and input/output ports, or they can have dedicated blocks of memory and dedicated ports. One of the CPU boards exerts the main control of the bus and is called the *master* board. The others are known most commonly as *slave* boards and sometimes as *user* boards, *service* boards, *satellite* boards, etc. (by the way, memory and I/O boards are also classified as slave boards). In a multiuser system, each user can have his/her own dedicated slave board and access the master's resources. You see how this is very similar to a network that connects individual computers, the only difference being that here all the CPUs stay inside one box.

Currently the media is hyped about the new Commodore Amiga 2000 because it features two CPUs. As with a great part of other 'new' technology being produced today, multiple CPUs have been implemented in S-100 computers a long long time ago (by microcomputer standards of time, of course). S-100 micros that feature up to 20 CPUs inside the box are commonplace, and some OEMs have implemented a lot more.

I won't go into any more detail about the box network in this particular article because essentially that is what a great part of S-100 Journal is about. Everytime you read other articles or ads that mention masters and slaves, this is what we are referring to.

The advantage of the box network is that each user workstation needs only to be an inexpensive terminal, not a whole new computer, and LAN interface cards (we'll discuss these further below) are **not** required. A disadvantage is that each terminal connects to the S-100 via a somewhat expensive, usually 10-wire, serial cable. In many installations, a combination of box network and coax-cable network is the ideal solution (pages 30 and 31 of our last issue, S-100 Journal No. 4, show such a combination.)

THE COAX-CABLE NETWORK

I don't care for IBM-PCs and clones. Yet, even I have to recognize that in many instances it is more useful to use a PC connected to the S-100 computer than a regular terminal connected to the S-100. After all, the ►

TRANSACTION PROCESSING SYSTEMS

HIGH PERFORMANCE ● FAST ACCESS ● LOW COST

Action Computer Enterprise, Inc., pioneers in the field of S-100 multiprocessor technology since 1979, brings you:

THE TRANSACTION™

DATABASE MANAGEMENT MACHINE

The transaction processing system that's designed to break price/ performance barriers with unmatched micro-based speed and capacity.

- 64 users — local or remote
- 85 Mbyte hard disk — expandable to 340 Mbytes or more
- 60 Mbyte streaming tape and floppy drive
- 80186 Service Processor 8 MHz 1 Mbyte RAM
- 8086 8-user Application Processor 8 MHz 1 Mbyte RAM
- Caching controller option
- 500 transactions per minute in a 64-user environment with a million key database
- MULTIUSER TAS — 4th Generation Application Development Language
- DPC/OS-86 Multiuser Operating System
- Upgradability. Hardware expandable, source available for application customization and maintenance.
- 64-user system for under $50,000!

APPLICATION DEVELOPERS —

MULTIUSER TAS, with its transaction-specific qualities, is an industrial strength Fourth Generation Language that's easy to use, easy to port and dramatically cuts development time. The combination of TRANSACTION and MULTIUSER TAS allows you to move into large configuration/high performance demand areas previously unavailable to micro-based systems.

Action Computer has all the tools to help you develop affordable, powerhouse transaction processing systems.

Call today, at: 1-800-821-6596 • In California call: 818-351-5451

TRANSACTION and **DPC/OS** are trademarks of Action Computer Enterprise, Inc.
MULTIUSER TAS, the version for all **dpc/os** and 16-bit Digital Research compatible operating systems, is licensed by Action Computer Enterprise, Inc. under exclusive agreement from Business Tools, Inc.

Action Computer Enterprise, Incorporated

430 North Halstead Street, Pasadena, CA 91107 USA TLX 279865 ACEI UR

prices of terminals and PCs overlap considerably. I feel therefore compelled to publish and write articles explaining this technology. Besides, there are far too many John Martins out there who have only now heard of the S-100 bus and who are prime candidates to join the S-100 community.

The coax-cable network is basically a number of computers connected by coaxial cable. In this section, I describe in detail the hardware components required to implement such a network with S-100s and PCs.

Hardware Components

All the hardware components required to set up a S-100/PC network are pictured on Photo 1, pages 30-31. The boards pictured are from Earth Computers. Boards from other manufacturers may look physically different and have cable distances somewhat different than those given below, but the functions are essentially the same. Here's a detailed description of the components pictured on Photo 1:

A. S-100 LAN Controller Board with Active Hub. This board provides the physical interface between the S-100 bus and the network. It is built around the COM9026 special-purpose communications chip from Standard Microsystems Corporation. The S-100 master accesses the LAN controller board through an I/O port address. The LAN controller retransmits data received from the master in a format that the LAN can understand (such as the ARCNET protocol) and also performs the reverse operation when receiving data. The small extension of yellow-and-black wire shown on the photo is to allow mounting the coax-cable female connector on the S-100 frame for easy coax-cable hook-up.

This particular board, A, also features an on-board 8-port **active hub**. Notice that there are nine connectors at the top, the original one plus eight from the hub.

B. S-100 LAN Controller Board. This is the same board as A, except that it has no active hub. The board can be purchased in either version.

C. 4-Port Active Hub. The active hub is optional and has one main purpose: to keep the signal (data) alive for longer-distance transmissions. A cable can be up to 2,000 feet long. After this distance, the signal becomes too weak to be recognized by the LAN interface boards. The hub circuitry picks up weak signals coming in any of the coax ports, amplifies them, and retransmits to the other ports. A serial line of alternating active hubs and 2,000-foot lengths of cable can extend up to a distance of 40 miles, allowing a very wide 'local' area network. The hub usually has several ports and thus also functions as a distribution center to increase the number of computers that can be connected to the network.

Both active hubs pictured are mounted on a computer card (A and C) for convenience, but this is not necessary. All an active hub requires is power so it is also available (not pictured) as a stand-alone small box that connects to a wall outlet.

Also not pictured are **passive hubs**. These are optional inexpensive units, requiring no power or circuitry, whose sole purpose is to fork the cable. Since a passive hub divides the signal from one cable into several without amplifying it, the intensity of the signal is actually degraded. This limits the cables connected to a passive hub to a number of four and to a length of 100 feet (200 feet if the other end is connected to an active hub). Furthermore, a passive hub cannot be connected to another passive hub.

D and E. PC LAN Controller Board. This is functionally the same as board B, except that it is built for the PC bus. Each PC that is connected to the LAN requires one LAN controller board.

F. Coaxial Cable. The cable that interconnects the computers and hubs is similar to the cable used in connecting VCRs to TVs. It is easily available and very inexpensive. To set up a test network, I bought 50 ft for under $10. Coax cable allows very fast speeds. For the boards pictured, the transmission rate is 2.5 Megabits per second.

Hardware Configurations

The number and manner in which the various computers are connected is very versatile and easily adaptable. Photo 2 represents the network in its simplest form. S-100 card A from Photo 1 is pictured but the hub is not being utilized, so card B would do just as well. In this network, a single PC is connected to a single S-100. Such a small network could be useful to a programmer wanting to use the two systems for software development while having an easy way to share the same files and peripherals. Or to a small business with a few user terminals and one PC workstation connected to the S-100. In either case, the PC only needs to be a very basic clone since hard disk, memory, and peripheral expansions can be done on the S-100 and accessed by both.

The network represented in ►

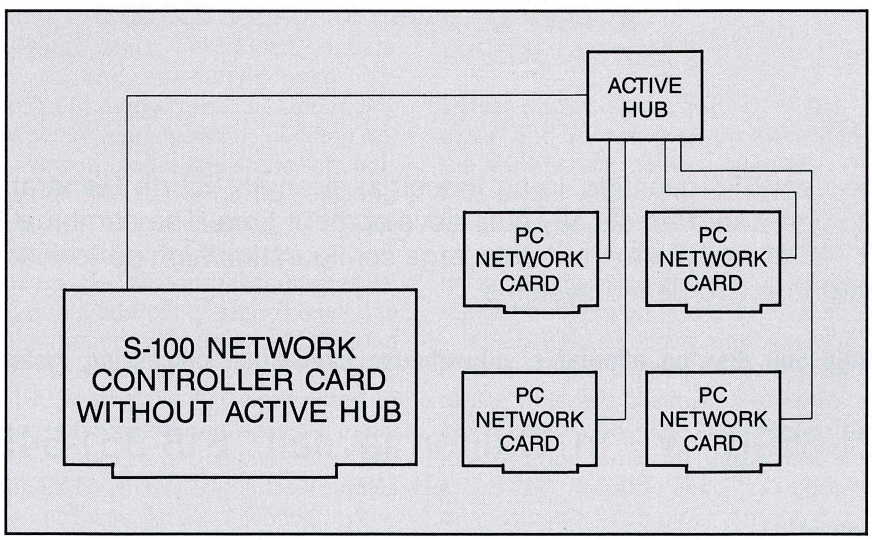


Figure 1. *In this network example, four PCs are connected to an S-100. The active hub allows the distance between PCs and S-100 to be up to 4,000 feet.*

Meet me at the FAIRE

THE 12TH WEST COAST COMPUTER FAIRE

MARCH 26–29, 1987 • MOSCONE CENTER • SAN FRANCISCO

The Faire That Has Everyone There!!!

Computer users, business people, hobbyists, enthusiasts! It's that time of year again! Time for the largest, longest running marketing event in the microcomputer industry. The West Coast Computer Faire! And anyone who's into computers is sure to be there!

Come on out and see the latest and greatest products from the world's leading companies. Plus lots of impressive innovations from new up and comers. And in keeping with the true tradition of the Faire, there'll be all kinds of special attractions like...

An extensive Conference Series, Professional Development Seminars, "Hands On" Demonstration Rooms, Computer Consultation Clinics, Drawings, Giveaways, Prizes, and much, much more!

Catch all the excitement of what's going on in the microcomputer industry today. Gather your computer pals together and tell them to meet you at the Faire!

THE 12TH WEST COAST COMPUTER FAIRE

Advance discount tickets available at:

BASS TICKETMASTER® Charge by phone (415) 762-BASS (408) 998-BASS.

Or call the Bass Outlet nearest you.

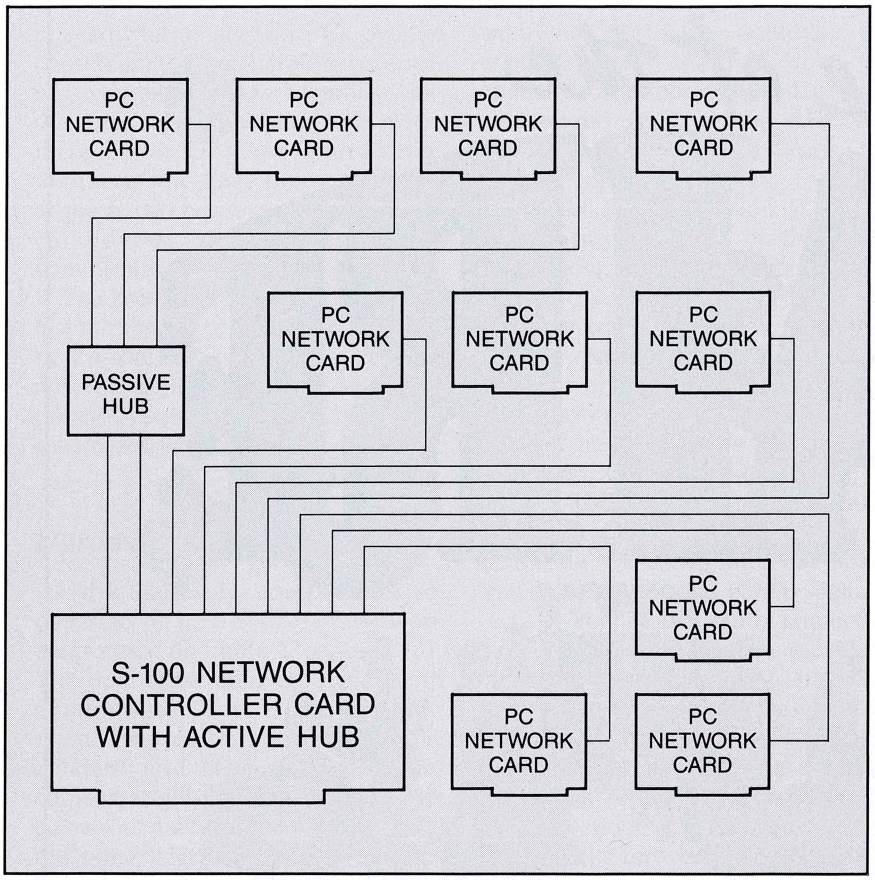


Figure 2. *In this network, ten PCs connect to the S-100 main system. The hub on the S-100 board plus the passive hub provide the necessary interface.*

Photo 1 connects two S-100s and two PCs. Several ports are available for further expansion through the hub on board A. Hub C could instead be a stand alone box, or it could be deleted and everything connected to the hub on board A. Having hub C or not would most likely be dictated by distance and the routing of the installation. With hub C installed in one of the PCs, the distance between any two of the three computers not housing hub C could be up to 4,000 feet. Many peripherals, such as hard disks and printers, could be connected to the system that houses board A and accessed by the other computers. Depending on the LAN software implemented and the purpose of each station (usually called a **node** in the network), peripherals connected to B, D, or E may or may not be accessed by the others. In addition, some processors could be set up as batch processors.

Figure 1 represents another installation using a stand-alone hub halfway between the S-100 server and the PC workstations. Such a configuration would be required if all the PCs had to be 2,000 to 4,000 ft away from the server.

In Figure 2, the active hub on board A and a passive hub are used to connect ten PCs to the S-100 server. The maximum cable distance between the server and the PCs connected to the passive hub is 300 ft. The cable distance to the other PCs could be up to 2,000 ft.

With the hardware described here, there is a maximum of 255 nodes (workstations or servers). Of course many S-100 nodes could be multiuser systems, allowing thousands of users on the network.

Each node (LAN controller card) on the network is assigned a node number for unique addressing. The node number for each card is selected via DIP switches on the card. In a typical network, each printer or disk drive is also assigned a number (or letter). A user on a workstation wanting to access a drive on the server would simply type the drive letter as if it were a local drive. In most implementations, the process of accessing a remote drive or printer is transparent to the user (there is no difference, from the user point of view, between accessing a remote and a local peripheral).

LAN SOFTWARE

To make a network function, a controlling operating system must be installed. The actual performance of a LAN is intimately tied to the software managing it. Earth Computers and other vendors of S-100 LAN hardware generally offer software to run their products. They can provide details on what they have to offer. Other S-100 Journal articles, such as Akin Orhun's article in issue No. 4 and others to come, will address S-100 LAN software issues more specifically.

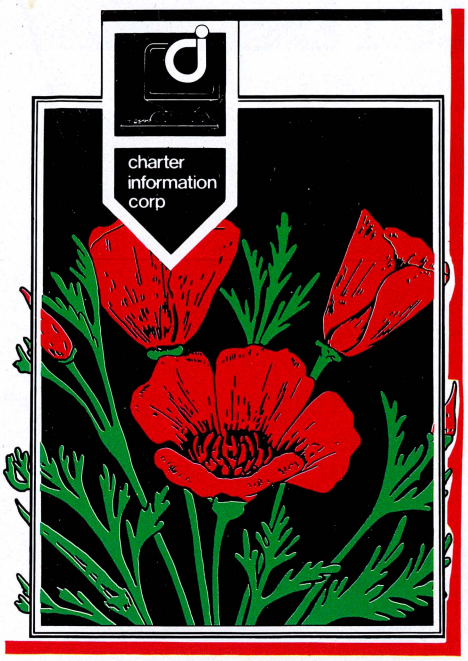
S-100 Journal thanks Earth Computers for technical support and for loaning us the hardware necessary to set up a test LAN. For more information on the products mentioned, please contact or send an Editorial Feature Reply Card to:

Earth Computers
PO Box 8067
Fountain Valley, CA 92728
(714) 964-5784

or contact Earth's S-100 distributor:

Lodden Technology Limited
2834 N. Milford Rd.
Milford, MI 48042
(313) 685-8538

COME VISIT THE
S-100 JOURNAL BOOTH
AT THE 12th WEST COAST
COMPUTER FAIRE
MARCH 26-29, 1987
Moscone Center, San Francisco
BOOTH 517



POPPY
brings compatibility to full flower.

Multi-User • PC-DOS • Turbo-DOS • S-100

Yes, they can co-exist!

And you'll love the way they look!

This remarkable workstation from Charter Information Corp gives you monochrome, color, and EGA capabilities all on one blooming board which fits into your multi-user Turbo-DOS system. In fact, you can fit up to sixteen POPPY boards inside a box for true multi-user capabilities. POPPY uses PC compatible monitors and keyboards which can be up to 200 feet from the central system.

Run all your favorite PC-style programs, including Lotus 1-2-3, Wordstar, Sidekick, Flight Simulator, Jet, and more under PC-DOS or Turbo-DOS on your S-100 system. And then share your peripheral resources among various users.

And if you're a reseller who hasn't made multi-user machines before, give us a call. We'll make the whole system for you in 3 to 10 days. If you missed seeing us at COMDEX, be sure to write or call for our free brochure. We think you'll agree that POPPY will really add to your computing landscape.

You've planted a seed in my mind.

☐ Please send me more information on the systems that you will integrate for me.

☐ Send information on being a dealer for your product.

Name _____

Title _____

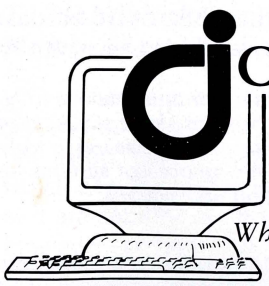
Company _____

Address _____

City _____

State _____ Zip _____

Telephone (___) _____



Charter Information Corp

Where good ideas improve.

2421 Rutland
Austin, Texas 78758
(512) 835-1111

DESIGNING A 1-MEGABYTE STATIC MEMORY BOARD

Kevin Parker

One unique feature of the S-100 bus that has enabled S-100 systems to thrive and prosper has been its ability to support higher performances while remaining compatible with older components. The bus has grown to accommodate multiple 8- and 16-bit microprocessors and sophisticated multiuser systems. And it is now expanding to include the new 32-bit ultra-high-speed processors (see 32-bit articles in S-100 Journal No. 4 and No. 5).

Despite the steady growth of the S-100 bus, it is not always easy to find good technical information on bus interfacing and bus operation. This article provides insight on implementing 16-bit memory and on interfacing to the S-100 bus in general. I will briefly review the 16-bit data transfer protocol, identify a couple of shortcomings in the IEEE-696 bus standard, and then show how I designed a 1-Megabyte 8/16-bit static RAM board. The 8/16-bit interface used on this board is suitable for use, with minor modifications, on essentially any 16-bit slave board.

This article describes how to design a 1-Megabyte, 8/16-bit, static memory board for the S-100 bus.

For those who do not wish to build S-100 hardware, the same memory board is available fully assembled and tested in various configurations. Please see Table 1 on page 40 for parts and ordering information.

QUICK REVIEW OF DATA TRANSFERS ON THE S-100 BUS

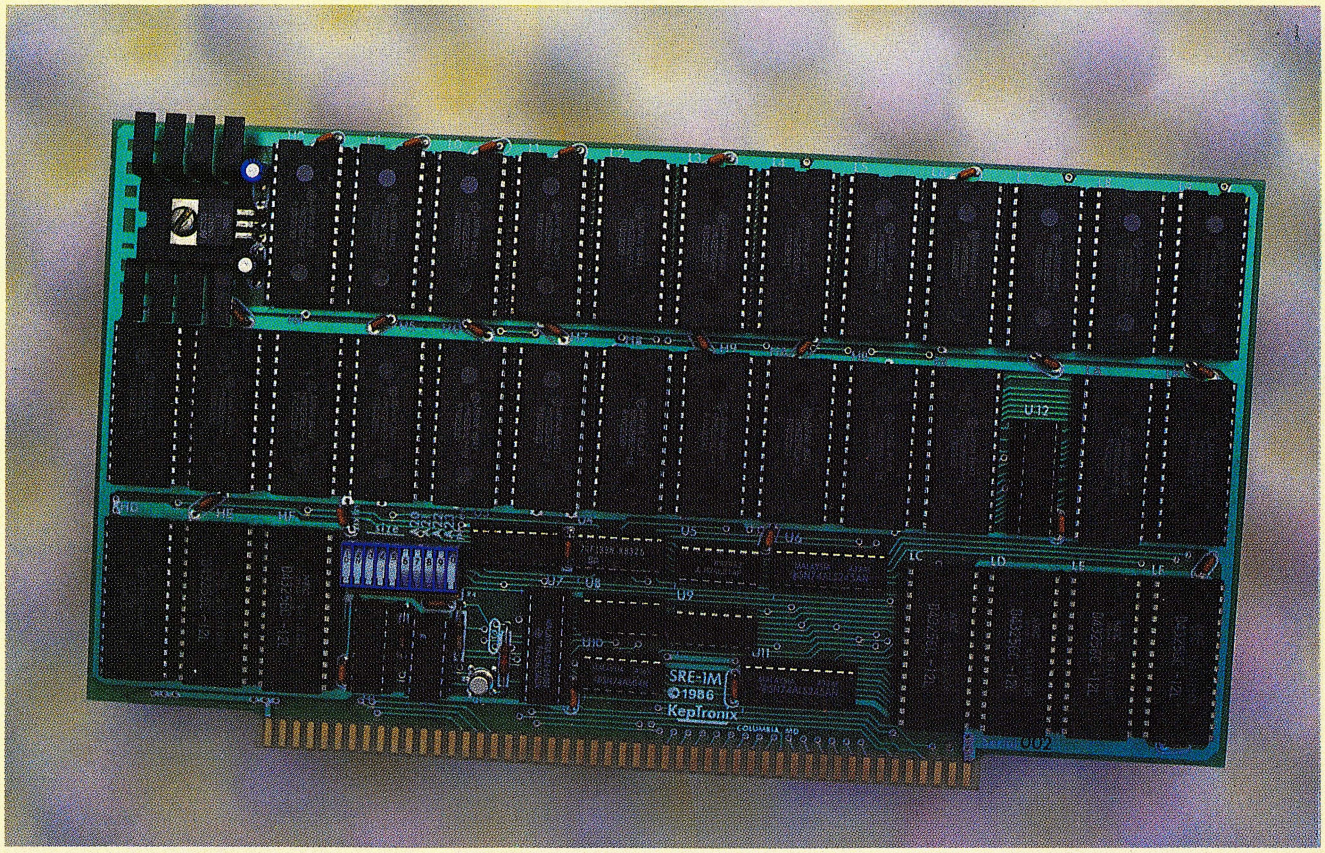
Before I explain the interface design, let's review how data is transferred between devices on the S-100 bus.

Byte (8-bit) data transfers take place over two separate unidirectional data buses (a bus is a set of signal lines on the motherboard). The S-100 **data input** bus is used to transfer —read— data from slaves (I/O interface and memory boards) to the bus master (CPU board, DMA controller, etc.) The **data output** bus is used to transfer —write— data from the bus master to a slave board. It takes one **bus cycle** for each data transfer. To determine if it should take part in a given bus cycle, each device (memory, I/O board, etc.) decodes the information present on the **address** and **status** buses. Two strobes, pDBIN and pWR* respectively, are used to gate (during a read cycle) and to validate (during a write cycle) data on the data bus. Figure 1 shows the relationship of these signals during memory read and write cycles. A bus cycle is composed of three bus states plus additional wait states if necessary; in Figure 1 wait states have been omitted for clarity.

Word (16-bit) data transfers are executed by combining the data input and data output buses into one bidirectional 16-bit data bus. During a word transfer, the data input bus carries the odd-addressed byte while the data output bus carries the even-addressed byte.

The S-100 bus uses two handshaking signals, sXTRQ* and SIXTN*, to specify whether 8-bit or 16-bit data will be transferred during any given ►

Kevin Parker is a Communications Engineer at Johns Hopkins University Applied Physics Laboratory. He specializes in high-performance microprocessor systems for signal processing applications.



bus cycle. When the bus master wants to perform a 16-bit transfer, it asserts sXTRQ* (SiXTeen ReQuest), in addition to driving the other status lines, during Bus State 1. If the addressed slave is capable of 16-bit operation, it must respond by asserting SIXTN* (SIXTeeN acknowledge) prior to the beginning of Bus State 2. If the addressed slave does not assert SIXTN* within the time constraints shown in Figure 1, the bus master will normally initiate two 8-bit bus cycles to effect the 16-bit transfer.

Since some bus masters, e.g., 8-bit CPUs and most DMA controllers, may need to access only 8 bits of a 16-bit word, all 16-bit devices must be capable of transferring either 8 or 16 bits at a time.

To Handshake or Not to Handshake

When should a bus slave drive the SIXTN* bus line? Clearly, slaves must not assert SIXTN* unless they are selected by the address and status bus and are capable of 16-bit operation. However, 16-bit slaves must also ensure that the SIXTN* line is asserted

1-MEGABYTE BOARD FEATURES

- The board can be populated with as little as 64K of memory; additional chips can be added in 64K (2-chip) increments as system requirements dictate.
- DIP-switch board size selection allows a partially populated board to be used with other memory boards residing in the same 1-Megabyte address space.
- Uses new high-speed, low-power 32K×8-bit static RAM chips.
- Runs at 10 MHz with no wait states.
- Low power consumption. Typically about 500 mA (fully populated).
- Compact. One single-height card requiring only 1 slot.
- Automatically handles 8 and 16-bit transfers (IEEE 696 compatible).
- 16-bit operation can be disabled for 8-bit-only systems and performance testing.
- Addressable on any 1-Meg boundary in the 16-Meg address space.

Qty.	ID	Part Description
1	U1	74F283 4-bit full adder
1	U2	74AS136 Quad X-OR gate
2	U3, U4	74F138 3x8 decoder
2	U5, U9	74LS14 Hex Schmitt trigger
2	U6, U11	74ALS245 Bidirectional bus transceivers
1	U7	KepTronix #KT861b (16L8A PAL)
1	U8	74F00 Quad 2-input NAND gate
1	U10	74AS04 Hex inverter
1	U12	74F245 Bidirectional bus transceiver
1	Q1	2N918 Transistor
1	VR1	LM7805 5-Volt regulator (TO-220 case)
2-32*		uPD43256C-12L or HM62256P-12L RAM
1	R4	10KΩ x9 resistor pack (SIP)
1	R2	10KΩ ¼-watt resistor
2	R1, R3	1KΩ ¼-watt resistor
1	C2	3.3 uF 20V (min) capacitor
1	C1	.47 uF 10V (min) capacitor
25		.1 uf. 20V (min) decoupling capacitor
5		14-pin IC socket
3		16-pin IC socket
5		20-pin IC socket
32		28-pin IC socket
1		Heatsink (TO-220 case)
1		DIP switch (10 position)
1		KepTronix SRE-1M printed circuit board

*Two RAM chips per 64K

The following items are available from KepTronix, PO Box 2022, Columbia, MD 21045, (301)-381-4297:

Technical manual only	$10.00
PC board, PAL chip, and technical manual	$150.00
Assembled and tested: 64K installed	$350.00
256K installed	$475.00
512K installed	$775.00
768K installed	$995.00
1 Megabyte installed	$1150.00

Assembled and tested boards are covered by a 1-year limited warranty (copy of warranty available on request). Personal check or money orders accepted. Please include $5.00 for shipping and handling. Add 50 cents per $100 insurance if desired. Maryland residents please add 5% sales tax.

Table 1. *List of parts required to build the 1-Megabyte static RAM board.*

70 nsec prior to the rising edge of Φ during Bus State 2.

After looking closely at Figure 1, you might conclude that the 70-nsec setup time required for SIXTN* cannot be met under any conditions because the bus status information, sXTRQ* in particular, is not guaranteed stable until 50 nsec prior to the rising edge of the bus clock (Φ) during Bus State 2. *(Editors note: a similar problem is also discussed in the 696 bus column in S-100 Journal Vol. 1 No. 4).*

To give slaves more time to respond to sXTRQ*, some bus masters circumvent this problem by sampling SIXTN* during Bus State 3 instead of Bus State 2, by providing status earlier in Bus State 1 than is required by the standard, or by sampling sXTRQ* later in Bus State 2. It is difficult to design sXTRQ*/SIXTN* handshake logic that will function properly with **all** bus masters because the approach taken by each manufacturer may be different. Unless you know the specific sXTRQ*/SIXTN* timing requirements for bus masters on your particular system, the best approach is to minimize the time required for the slave to respond to sXTRQ*. This requires using the fastest logic available for the SIXTN* generator circuit and minimizing address/status decode time on the board.

Another problem arises when 8-bit phantom slaves (selected when the S-100 signal PHANTOM* is low) and 16-bit non-phantom slaves (deselected when PHANTOM* is low) are located at the same address. Since, according to the standard, PHANTOM* is not required to be active until 30 nsec prior to pDBIN (or pWR*) becoming active, the bus master could sample the SIXTN* line **before** PHANTOM* is asserted. This means that a 16-bit non-phantom slave could erroneously drive SIXTN* low during a bus cycle that actually accessed an 8-bit phantom slave. If the bus master attempted a 16-bit bus cycle, the transfer would be invalid.

One solution to this problem is to ensure that bus masters do not request 16-bit transfers when PHANTOM* is active.

A better solution, that would also solve the sXTRQ*/SIXTN* timing problem discussed earlier, is to require bus masters to sample SIXTN*

during Bus State 3 instead of Bus State 2. I would recommend that this be incorporated into the S-100 bus standard. In the meantime, I strongly suggest that you follow this convention should you decide to design a bus master.

1-MEGABYTE STATIC RAM

The discussion of how I designed a 1-Megabyte static RAM board will illustrate 16-bit memory design techniques and provide the reader some insight into interfacing to the S-100 bus in general.

We must first establish a design specification. This is normally broken down into cost, density, and speed goals. Specifying performance goals and then designing to achieve them, however, is not as straightforward as it may seem. Achieving one goal invariably requires compromising the other two.

The type of memory to be used must also be considered. Static RAM allows the simplest and fastest designs, although it costs considerably more than dynamic RAM. Had dynamic RAM been selected, more bytes could fit on the board, but also more complex support circuitry would be required. Also, dynamic RAM is typically slower due to memory refresh requirements.

Before designing the board, I established the following design goals:

1. 1 Megabyte on one single-height board.

2. Low power consumption.

3. Fast. Operate at 10 MHz with no wait states.

4. 8 and 16-bit operation (of course!).

5. Addressable on any 1-Megabyte boundary.

6. When partially populated, operate with other memory boards residing in the same 1-Megabyte address space.

The first two design goals require that package count be held to an absolute minimum. The third goal requires that propagation delay (time required for an input to change a gate's output) be minimized in critical signal paths; this can be accomplished through reduced gate count and RAM chip selection. The selection of TTL logic family (74LSxx vs. 74Fxx, etc.) will be governed by both S-100 bus timing (AC) and bus loading (DC) specifications to ensure compatibility with all S-100 systems.

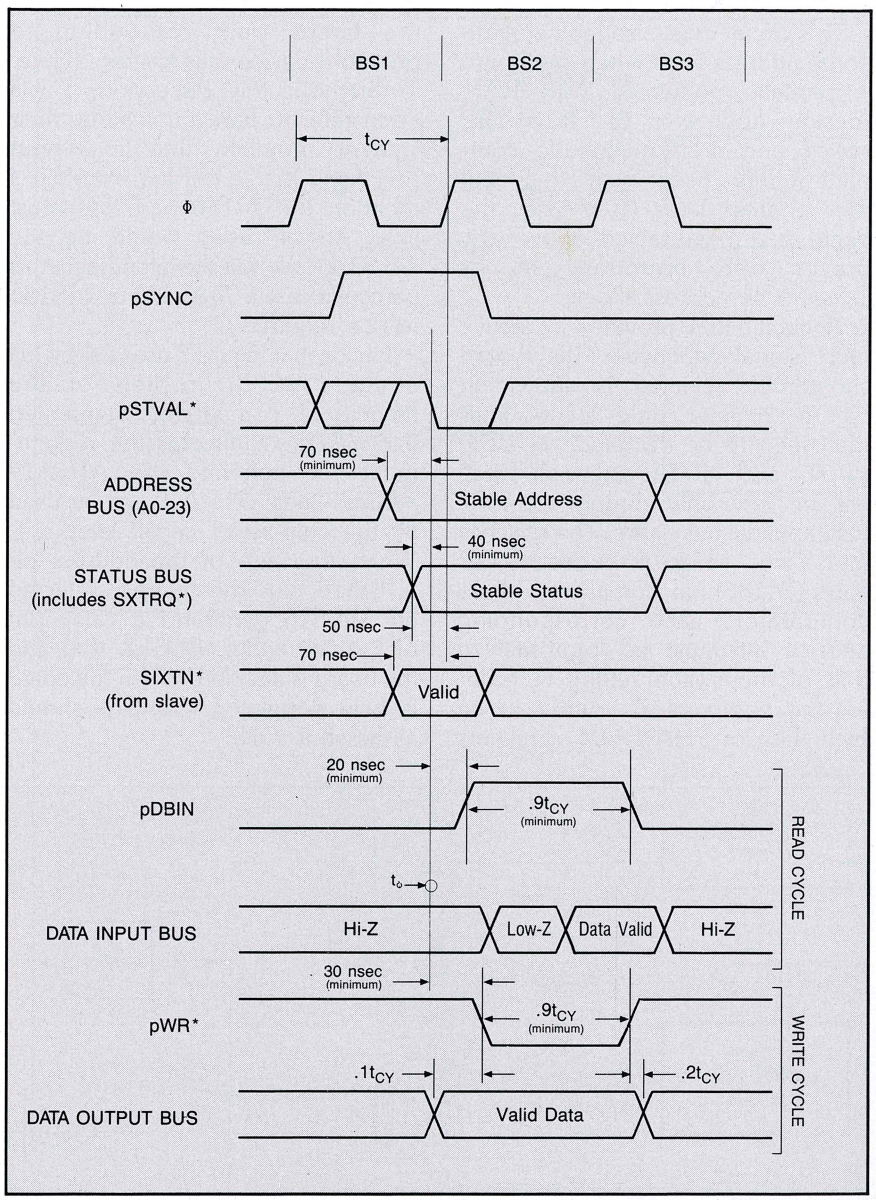


Figure 1. *S-100 bus-cycle and timing relationships.*

The sixth goal has been included to allow partially populating the board with RAM and thus keep initial costs down. Chips can later be added up to the full 1-Megabyte capacity.

The 32Kx8-bit static RAM chip was selected for this design because it is the only chip available (as far as density is concerned) that could meet the memory size goal. A total of 32 chips are required for 1 Megabyte of RAM.

It is easiest to break down the design into several smaller circuits. These are examined next according to their functions. Later I will show that some of the circuits can be simplified by using a PAL (Programmable Array Logic).

BOARD-SELECT CIRCUIT

To ensure that the board is only active during bus cycles intended to access the board's 1-Megabyte address space, we must decode the appropriate address and status bus lines. Since this board contains 1 Megabyte and the S-100 address bus can directly access up to 16 Megabytes, the top four address lines (A20-A23) must be decoded to generate the board-select signal.

This circuit must also look at additional address lines when the board is partially populated with RAM (design goal No. 6). If a DIP switch and 4-bit magnitude comparator are used with the next four address lines (A16-A19), the board can be disabled above any preset 64K boundary in its 1-Megabyte address space.

A circuit that provides an active high signal whenever the board should be selected is shown in Figure 2. The only status lines that need to be decoded are sINP, sOUT, and sINTA since all three will be low only during memory, halt, and idle bus cycles. The switches and open collector exclusive OR gates (74136) function as an address comparator; each corresponding address bit must be 'opposite' to that of the switch setting in order for the comparator's output to be high. Use of PHANTOM* prevents the board from responding to phantom-device bus cycles. (Note: Throughout this discussion, I will often refer to part numbers by their 'generic' number, that is without specifying the series. For example, I mention the 74136 part, but most likely to be used would be the 74**LS**136. We will see later that in this particular case a 74**AS**136 or 74**S**136 will be required.)

Notice that Figure 2 uses a 4-bit full adder (74283) in lieu of the magnitude comparator mentioned earlier. This eliminates the requirement for buffering the A16-A19 address lines (A16-A19 will be used by the chip-select circuit later on). When the sum of the address on A16-A19 plus the number selected by S1e-S1h generates a carry out of the full adder, BDSEL is 0 and the board is disabled. When the board is fully populated, S1e-S1h should all be shut (ON).

DATA BUS INTERFACE

To implement the S-100 data bus dynamic bus sizing protocol, the memory board is organized as two 512K×8-bit banks, referred to as EVEN and ODD. These are accessed either sequentially or simultaneously, depending on the type of memory transfer to be executed. The ODD bank is active during 8-bit odd-addressed (A0=1) byte reads and writes. The EVEN bank is active during 8-bit even-addressed (A0=0) reads and writes. Both banks are accessed during 16-bit transfers.

There are several ways to connect buffers together to interface the board data buses to the S-100 data bus. Two of the more popular schemes are shown in Figures 3 and 4. Bidirectional bus transceivers (74245) are used to minimize chip count. The circuit of Figure 3 is faster (less worst-case propagation delay) than that of

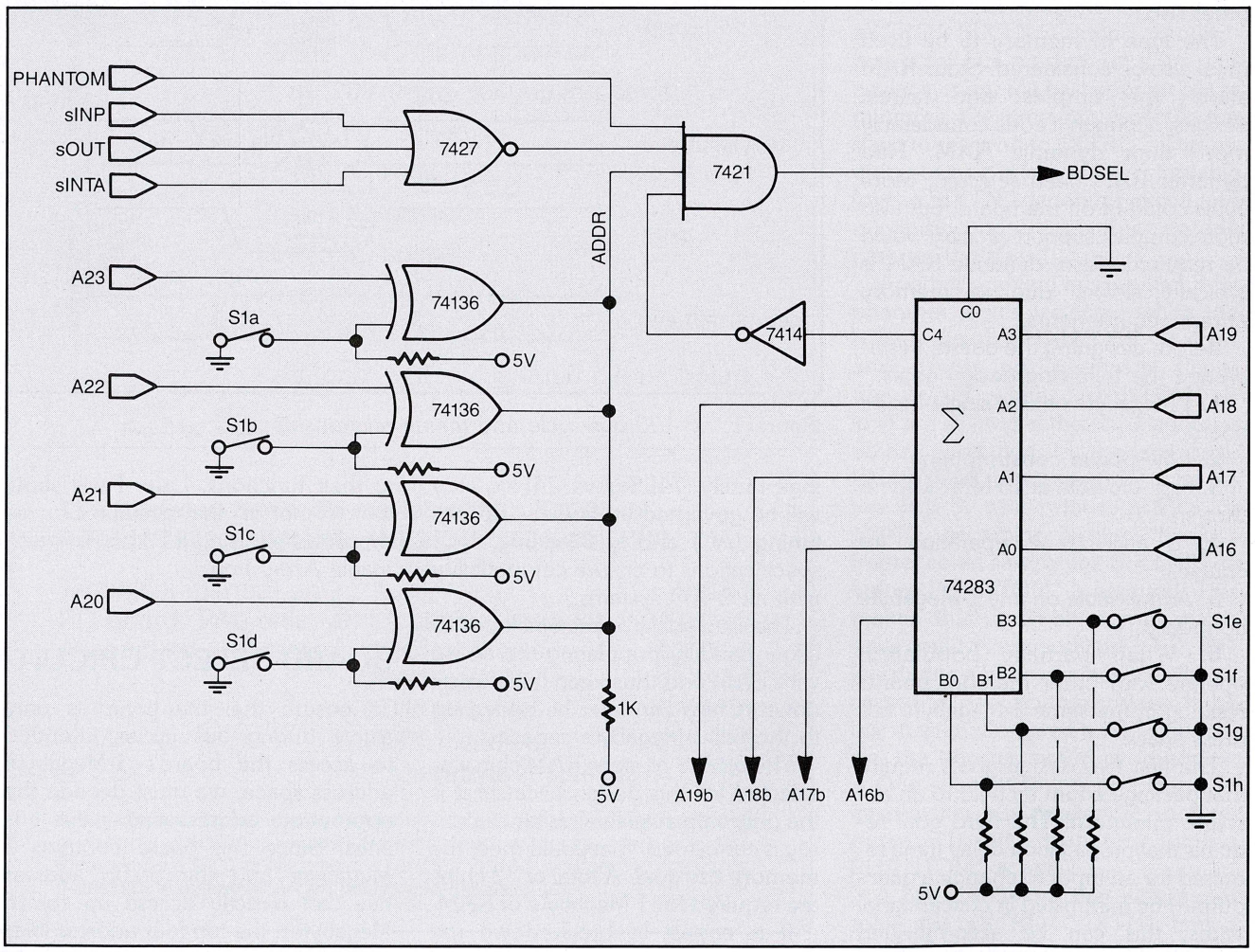


Figure 2. *The board-select circuit.*

Figure 4 because there is only one gate between the S-100 bus and the memory array for all possible data flow paths. Figure 4, however, requires one less chip. Since board space is at a premium, this last buffer configuration will be used.

Next we must design logic that will turn the correct buffers ON and OFF at the correct time. This will ensure that data is 'steered' in the right direction, between the appropriate memory board data bus and the S-100 data bus, according to the type of memory access currently in progress (e.g., read or write, 8 or 16-bit). Table 2 lists the buffers in Figure 4 that must be active for all (six) types of memory accesses. The absence of an 'x' denotes that the buffer output is in its high-impedance state for the specified bus cycle.

To define the six types of memory bus cycles, we must decode A0, sMEMR, and sXTRQ*. Since we are using 74245 bus transceivers for the data bus buffers, data-direction and transceiver-enable signals must be generated from this decoded status information. The 74245s' DIR lines may be connected directly to sMEMR since the card will drive the S-100 data bus only during memory read cycles. To prevent contention (two outputs attempting to drive the same line at the same time) between the board and the S-100 data bus, and between the RAM chips and the 74245s on the board, all transceiver EN (enable) lines are gated with bus strobes pDBIN and pWR*.

After considering all of the above, we can write the truth table (Table 3) for the bus transceiver control circuit. Figure 5 shows a circuit that meets the truth table conditions. Notice that the bus transceiver-enable lines are gated with BDSEL (Figure 2) so that the board does not attempt to drive the data bus when it is not selected. All S-100 bus lines are buffered to clean-up signals and minimize bus loading.

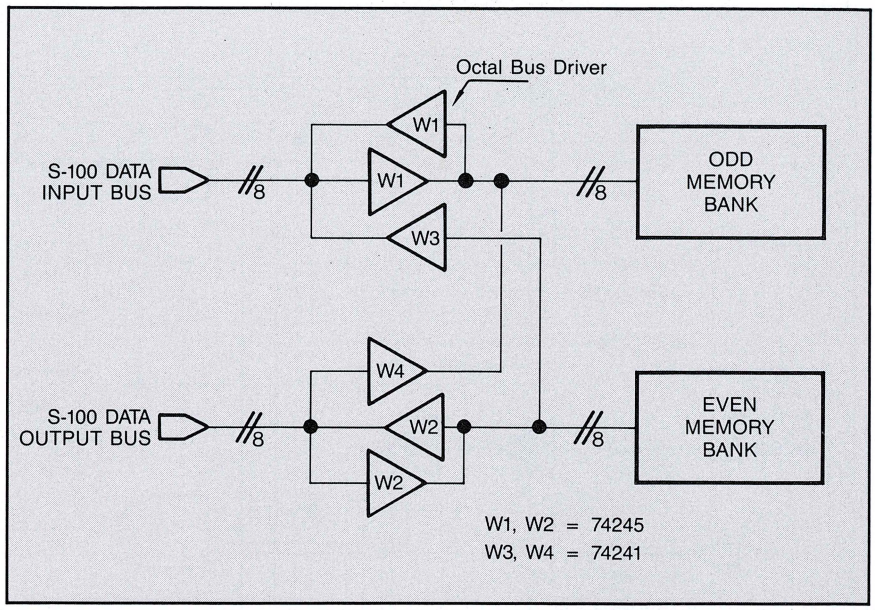


Figure 3. *Data bus interface. This circuit is faster but requires four chips.*

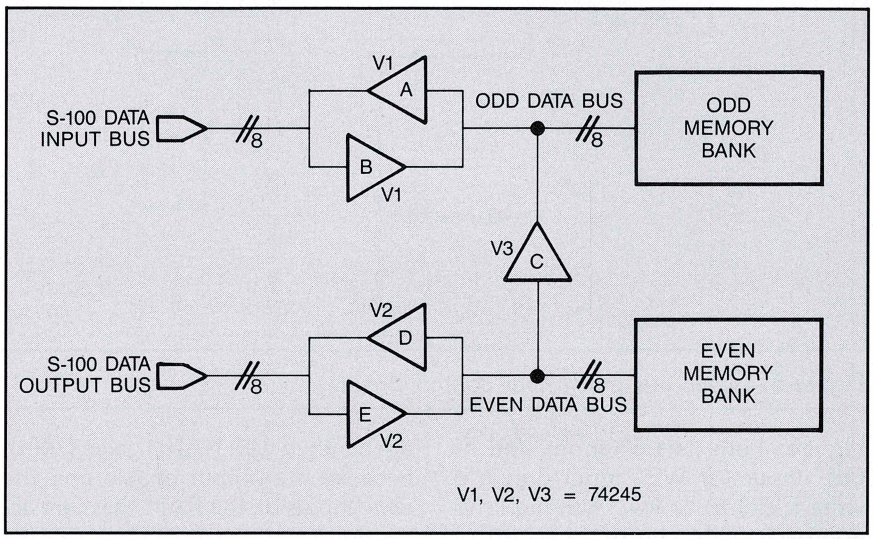


Figure 4. *Data bus interface. Although slower, this circuit is preferred because it only requires three chips.*

RAM CHIP AND BANK SELECTION

Individual RAM chips are selected by decoding the address bus during bus cycles in which the board is enabled (BDSEL=1). Each address should select a chip in both the EVEN and ODD memory banks since both banks will be accessed during 16-bit transfers. The A0 address line is used to choose between one bank or the other during 8-bit memory cycles.

Since there are sixteen RAM chips per bank, four address lines are necessary to generate individual chip-select (CS) signals. Addresses A20-A23 were used to generate BDSEL (Figure 2) and we reserve fifteen address lines (A1-A15) for selecting a unique memory location in each RAM chip. This leaves the four address lines A16-A19 to select individual chips.

One way to generate sixteen chip-select lines for each bank is by means of two 4×16 decoders — one for the EVEN bank and one for the ODD bank. A0 and sXTRQ* decoding circuitry would be connected to the address decoders' enable lines to select/deselect each bank for the various types of bus cycles. Unfortunately, 4×16 decoders come in standard 24-pin packages; using two of these chips would require too much board space.

Only one address decoder is required if A0 and sXTRQ* are used to gate the read and write strobes (pDBIN and pWR*) to each bank. This is possible because, to initiate a read (or write) cycle, the RAM chip

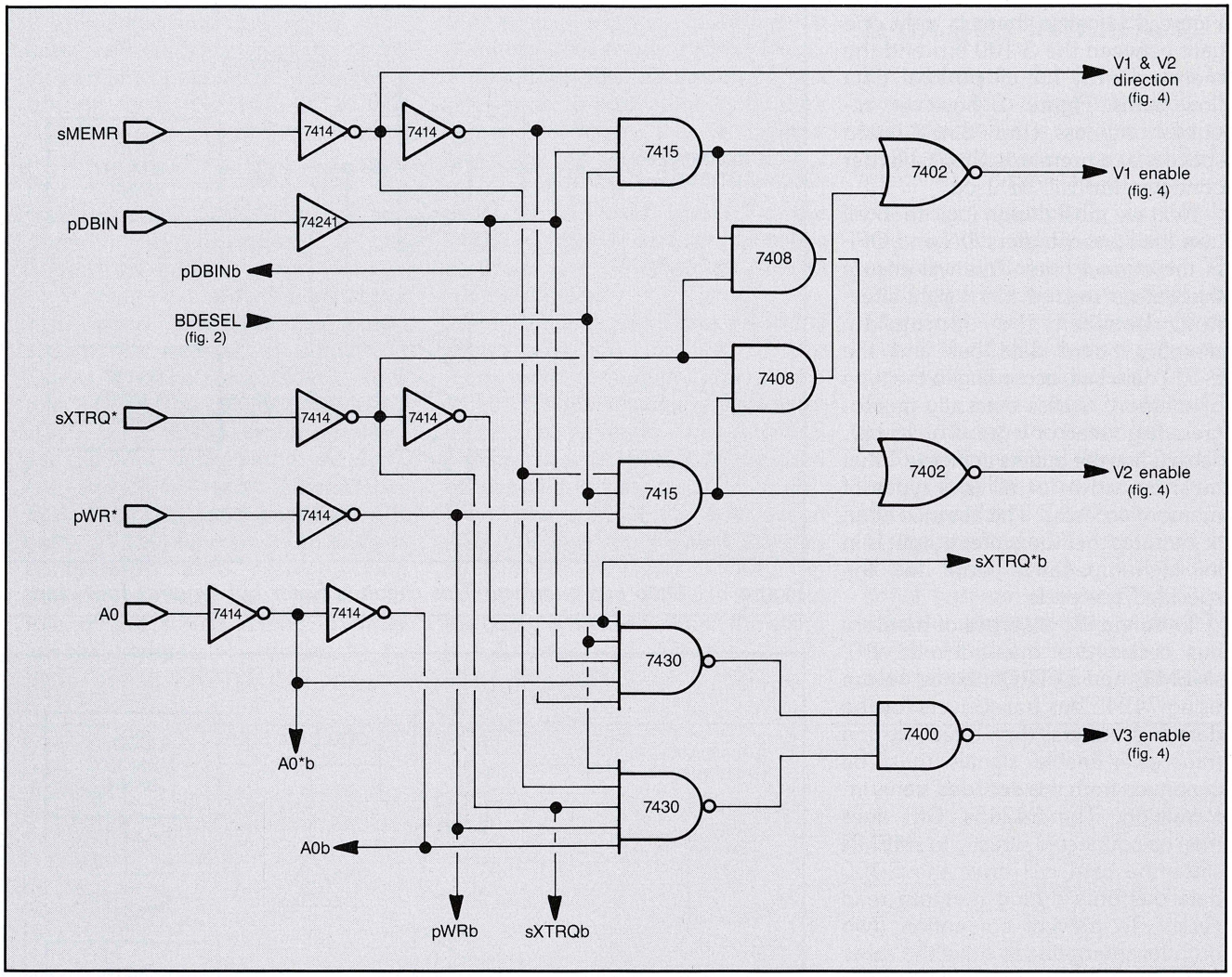


Figure 5. *Data bus transceiver control circuit.*

requires both its CS* input **and** its OE* input (or WE* input during a write cycle) to be low. Although this approach adds one gate's worth of propagation delay in the paths of the read and write strobe signals, the degradation will not be limiting if we use fast logic.

Figure 6 shows a circuit that selects individual RAM chips in both banks. I chose two 3×8 decoders (74138) instead of one 4×16 decoder (74154) for three reasons. First, the 74138 is available in the faster logic families. Second, the 74138 has both active high and active low enable inputs, simplifying the chip-select circuit design. And third, two 74138s require less board space than one 74154.

The AND (7408) and OR (7432) gates decode sXTRQ* and A0 to determine which bank(s) will receive the read (or write) strobe. The strobes are buffered with NAND gates (7400) because the output enable and the write inputs on the RAM chips are active low.

The S-100 bus MWRT signal was not implemented in this design to keep the circuit simple and minimize propagation delay. This will prevent the 'memory deposit' function of older front panels from working correctly with this board. In any case, it is unlikely that these front panels could access the full one megabyte address space of this board.

The schematic diagram for the

TYPE OF BUS CYCLE	ACTIVE BUFFERS				
	A	B	C	D	E
Byte read (even address)	X		X		
Byte read (odd address)	X				
Word read	X			X	
Byte write (even address)					X
Byte write (odd address)			X		X
Word write		X			X

Table 2. *Buffer activity versus type of S-100 bus cycle. No X indicates that the buffer output is in a high-impedance state.*

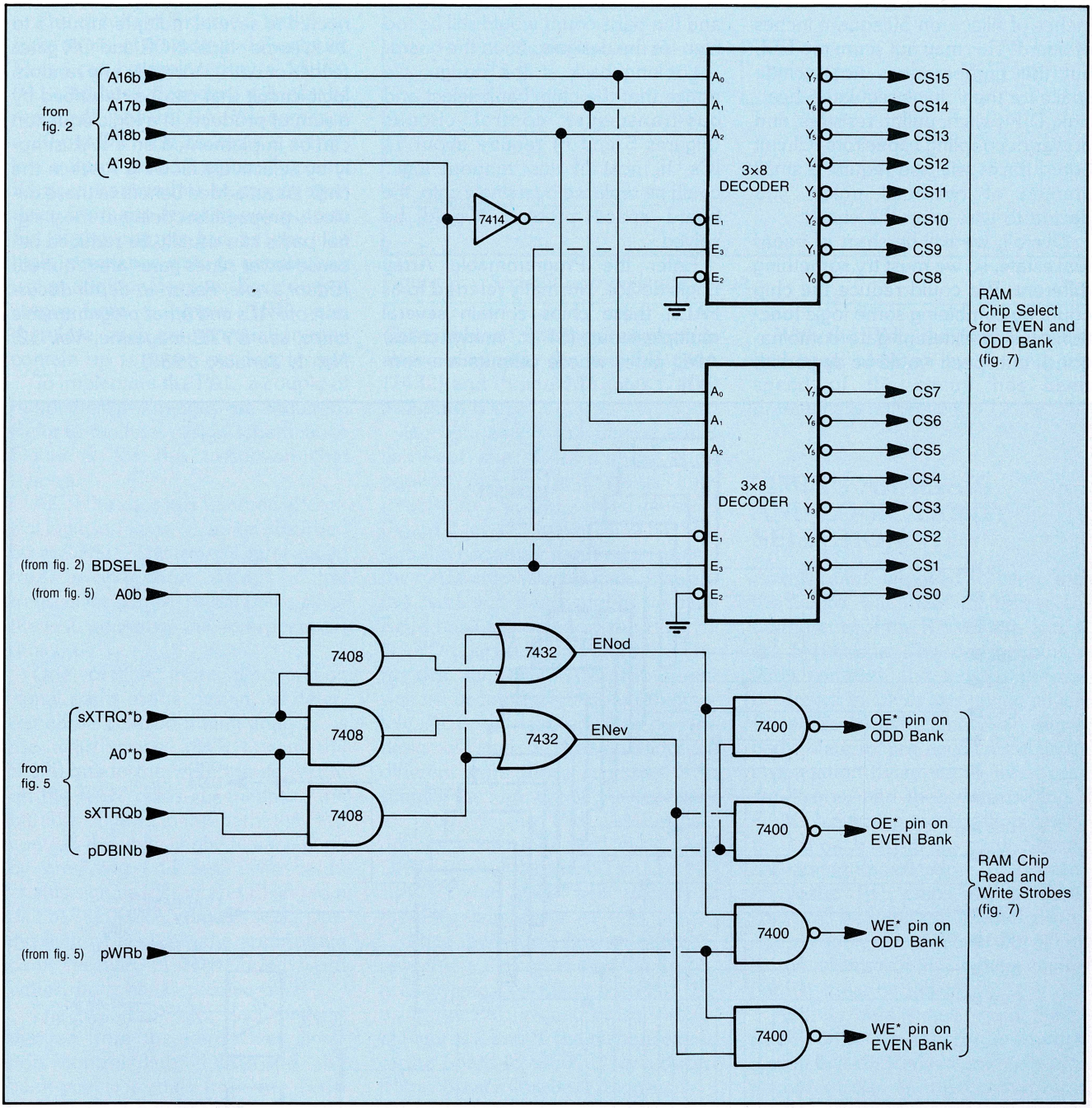


Figure 6. *RAM chip-select and bank-select circuit.*

EVEN and ODD memory banks is shown in Figure 7. Noninverting buffers (74LS241) are used on all address lines to minimize S-100 address bus loading. The DC bus loading specification could still be met without these buffers since the RAM chips are CMOS, but each address line would see about 180 pF of capacitance. This would exceed the bus standard specification (25 pF maximum) and could significantly degrade system performance or reliability.

16-BIT CYCLE ACKNOWLEDGE

The 16-bit handshake circuitry must pull the SIXTN* line low whenever the board is selected and a 16-bit cycle is requested. As discussed earlier, this circuit must be as fast as possible.

A NAND gate is about the simplest and fastest circuit possible. See Figure 8. An open collector gate must be used because SIXTN* is specified as an open collector signal in the S-100 bus standard. This circuit will not, however, assert SIXTN* soon enough, during phantom-device bus cycles, if the bus master samples SIXTN* before PHANTOM* is asserted.

PAL — PROGRAMMABLE ARRAY LOGIC

Before analyzing board timing, we calculate the space required by all the TTL and RAM chips and find that we are trying to cram about 42 square

inches of silicon on 50 square inches of board! This may not seem too bad, but the number does not include space for the voltage regulator, heatsink, DIP switch, pullup resistors, and power decoupling capacitors. Circuit board traces will also require a small amount of overhead unless you decide to wire-wrap the circuit.

Overall, we will be short of board real estate, so we must try something different. We could reduce the chip count by combining some logic functions (using different gate combinations), but speed would be degraded, and the parts count would still be too high for the design to fit on the board.

Looking back at the design, we notice that the chip/bank-select and bus-transceiver control circuits (Figures 5 and 6) require about 11 ICs. If most of this 'random logic' could be replaced by a single chip, the board space problem would be solved.

Enter the Programmable Array Logic device. Normally referred to as PALs, these chips contain several multiple-input (31 in many cases) AND gates whose outputs are connected to several multiple-input (8 to 16 in some chips) NOR and OR gates (either or both). Virtually any random logic circuit that can be described by a sum of products in a logic equation can be implemented on a PAL simply by selectively blowing fuses in the chip. As an added benefit of these devices, propagation delay in most signal paths can usually be reduced because fewer series gates are required. *(Editor's note: For an in-depth discussion of PALs and other programmable chips, see BYTE magazine, Vol. 12, No. 1, January 1987).*

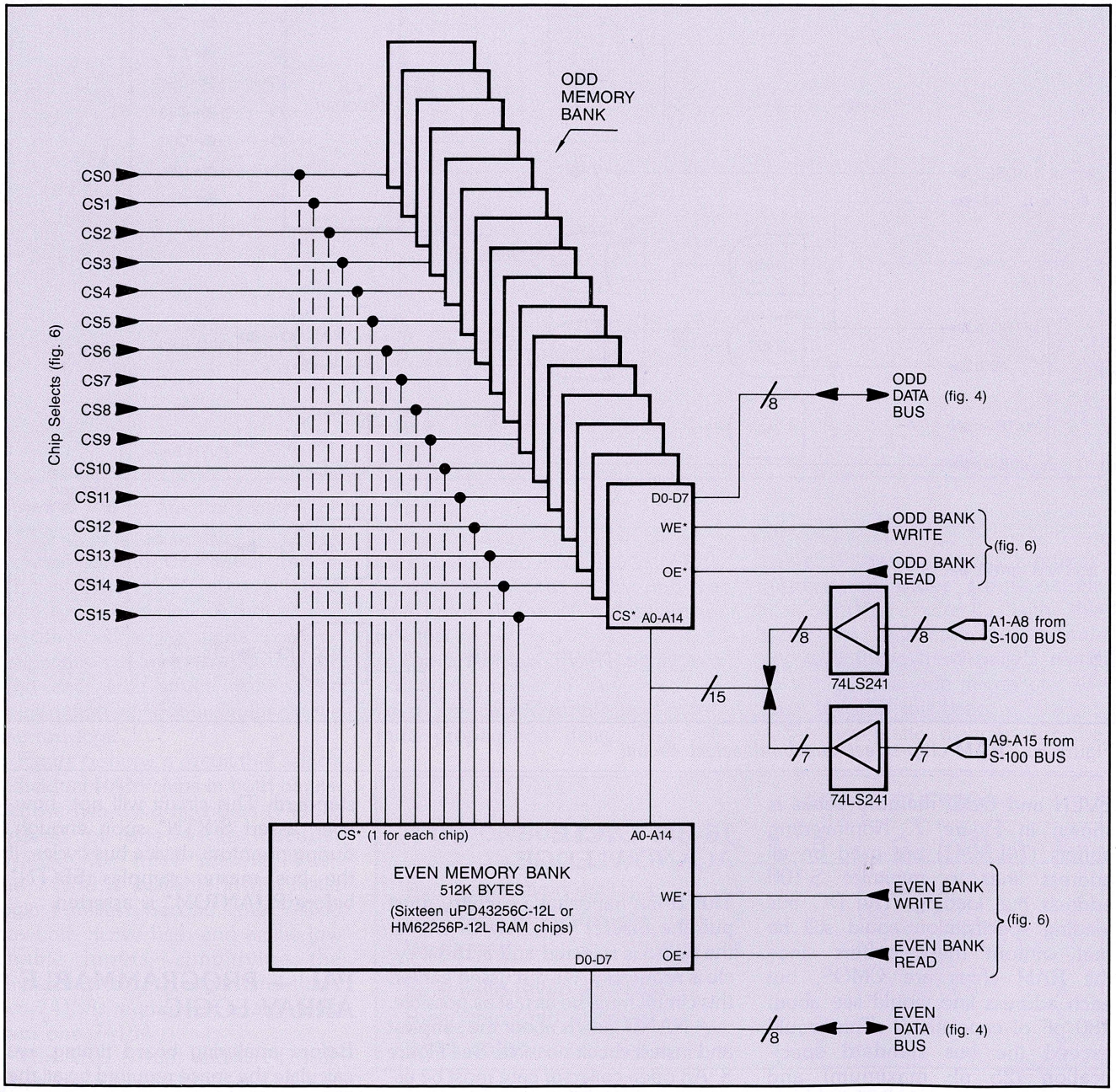


Figure 7. *Organization of static RAM chips on the 1-Megabyte RAM board.*

The circuits of Figures 5 and 6 require a total of 6 inputs and 8 outputs. These can be replaced by a 16L8A PAL (address decoders will not be replaced). The 16L8A PAL comes in a 20-pin DIP and contains 8 active-low outputs and 10 inputs. It inserts 25 nsec (maximum) of propagation delay from any input to any output. Six of the outputs can be put in a high-impedance state to permit their use as additional inputs. Each output comes from a 7-input NOR gate; therefore, each output equation may contain up to seven product terms.

To implement the PAL, a couple of minor design changes are required. Refer to the final design schematic in Figure 9 for the discussion that follows.

All of the data bus transceiver control logic in Figure 5 can be absorbed by the PAL. This results in reduced total propagation delay in the transceiver control signal paths since the PAL generates the control signals directly.

One of the more time-limiting signal paths in the design, as determined by detailed timing analysis, is the write strobe (pWR*) from the S-100 bus to the write inputs (WE*) on the RAM chips. To minimize the number of gates in this path, the PAL controls the write strobe to each bank by providing odd and even bank-enable signals (ENod and ENev) to a pair of 74F00 NAND gates (U8). During byte writes, only the appropriate bank receives pWR*. During word writes, both banks receive pWR*.

The read strobe for each bank is derived from the transceiver direction control line (DIR) and the bank-enable signals from the PAL.

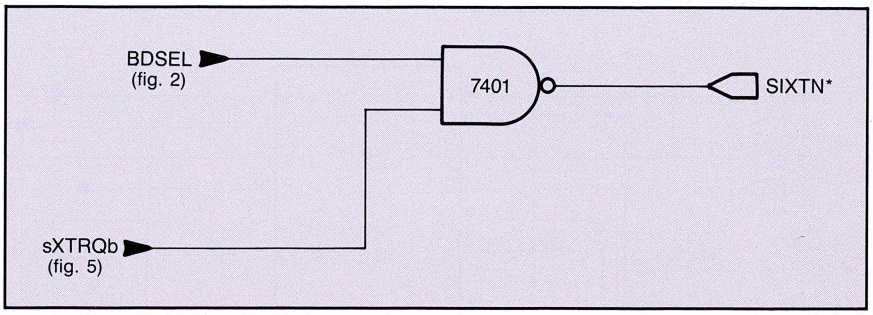


Figure 8. *Sixteen-acknowledge circuit.*

This combined with the change made above eliminates the two OR gates (7432) and three AND gates (7408) in Figure 6.

By connecting the binary adder carry-out line (C4), address comparator output, and status lines directly to the PAL, three gates in Figure 2 are eliminated. The BDSEL signal is no longer required to qualify the RAM-chip-select signals. Instead, the read and write strobes to each RAM bank are gated by the PAL via NAND gates (U8 on Figure 9). This results in the RAM-chip-select signals becoming valid approximately 40 nsec earlier than in the design of Figure 6. The status lines are different from those in Figure 2 to simplify the logic equations necessary to program the PAL; sMEMR=1 defines a memory read cycle (and DIR=0) while sOUT=0 AND sWO=1 (or sWO*=0) defines a memory write cycle.

Since there are a few unused pins remaining, we can absorb the 16-bit-acknowledge circuit in the PAL. By using the PAL instead of the circuits of Figures 2 and 8, the response time of the board to sXTRQ* is reduced from 30 nsec to about 15 nsec.

With the PAL, the total IC count has been reduced by 7, and the speed of the circuit has been dramatically improved.

TIMING ANALYSIS AND LOGIC FAMILY SELECTION

At this point, we need to verify that the board will meet all timing requirements of the RAM chips and of the S-100 bus. This can become a fairly complex and tedious process because it involves identifying critical (most restrictive timewise) signal paths, determining each critical path's propagation delay under worst-case conditions, and then determining if RAM-chip timing specifications can be met for each S-100 bus cycle. Should any timing specifications be exceeded, faster TTL chips and/or RAM chips will be required. If faster silicon doesn't solve the problem, the circuit in the vicinity of the critical timing path(s) must be redesigned.

Address/data/status setup times, RAM chip timing requirements, system bus clock speed, and gate propagation delay are factors that must

TYPE OF BUS CYCLE	S-100 BUS SIGNALS				74245 CONTROL SIGNALS					
					V1		V2		V3	
	sMEMR	A0	sXTRQ*	BDSEL	DIR	EN	DIR	EN	DIR	EN
Byte read (even address)	1	0	1	1	0	0	0	1	0	0
Byte read (odd address)	1	1	1	1	0	0	0	1	0	1
Word read	1	0	0	1	0	0	0	0	0	1
Byte write (even address)	0	0	1	1	1	1	1	0	0	1
Byte write (odd address)	0	1	1	1	1	1	1	0	0	0
Word write	0	0	0	1	1	0	1	0	0	1

Table 3. *Truth table for the data bus transceiver-control circuit.*

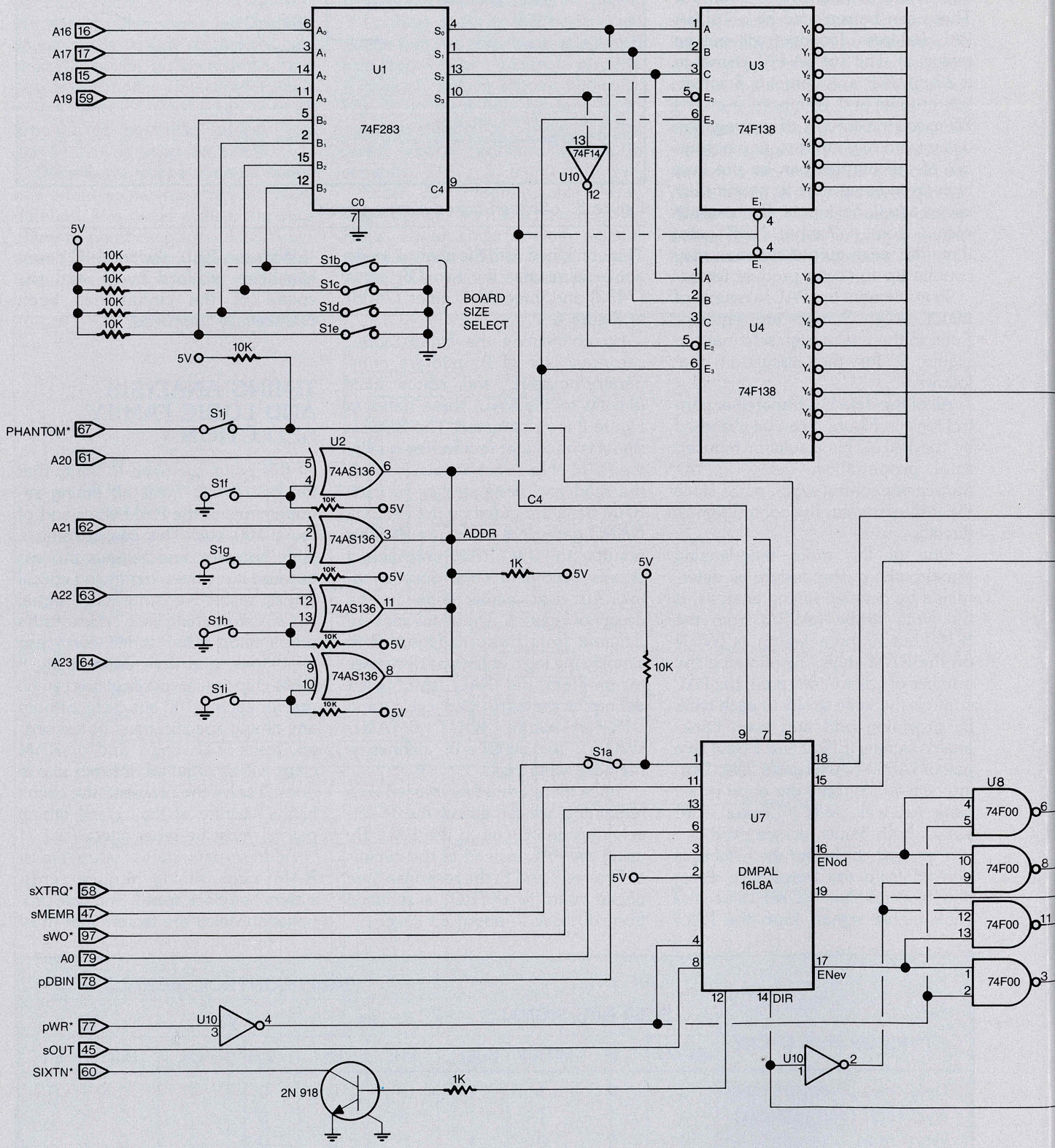
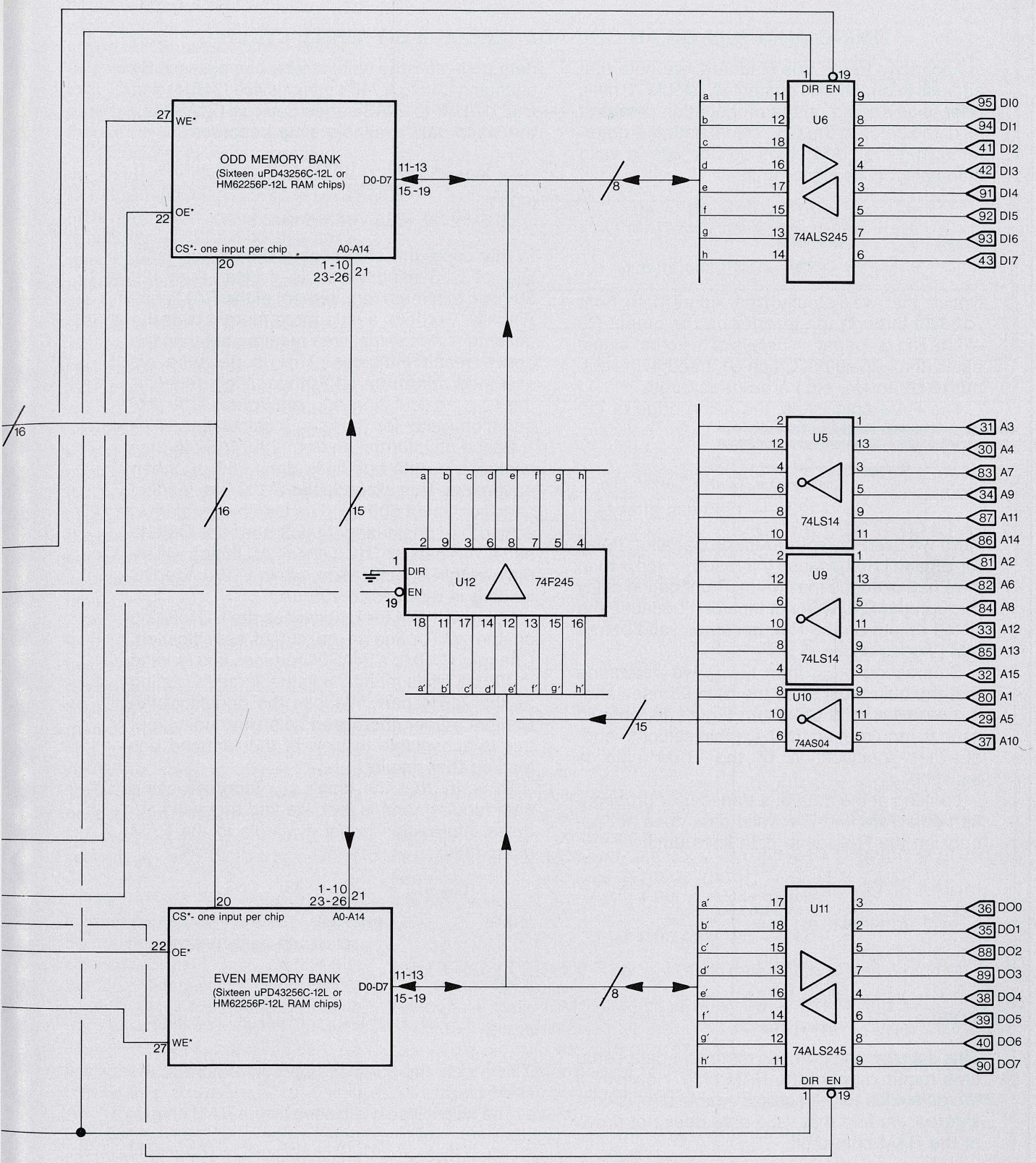


Figure 9. *The final design of the 1-Megabyte Static RAM board*



Memory board design by Kevin Parker

TIMING ANALYSIS OF AN ODD-ADDRESSED 8-BIT WRITE CYCLE

According to the bus standard, we note that address information is guaranteed stable 70 nsec prior to t_0. Adding up the propagation delays of all of the gates in the CS* signal path, we determine that the RAM chips do not 'see' a valid chip-select (CS) signal until:

(EQ A)
$$t_{CSval} = \underset{\text{(S-100 bus addr valid)}}{-70} + \underset{\text{(U1)}}{30} + \underset{\text{(U10)}}{33} + \underset{\text{(U4)}}{40}$$
$$= +33 \longrightarrow 33 \text{ nsec after } t_0$$

Notice that we selected the signal path from A16-A19 through the inverter on the output S3 of U1. The propagation delays of the other signal paths (through output C4 on U1, through the output of U2 to U4, etc.) are not as large.

The RAM chip 'sees' the rising edge of the write signal (WR*) at:

(EQ B)
$$t_{WRval} = \underset{\text{(trailing edge of pWR*)}}{[+30 + .9(100)]} + \underset{\text{(U10)}}{5} + \underset{\text{(U8)}}{3}$$
$$= +128 \longrightarrow 128 \text{ nsec after } t_0$$

Here we used the **minimum** propagation delay for U8 and U10 because the result is worse than if we had used the maximum. Propagation delay through the PAL was not included because the ENod output on the PAL becomes valid before pWR* goes active.

Subtracting t_{CSval} from t_{WRval}, we determine that the chip must store the data 95 nsec after it receives a valid CS. From Figure 10, only 85 nsec is required, so the t_{CW} (chip select-to-end-of-write) specification of the RAM chip is satisfied.

Looking at the data bus transceiver propagation delay, we find that valid data does not appear on the RAM chip data lines until:

(EQ C)
$$t_{DATAval} = \underset{\text{(data available on S-100 bus)}}{[+30 - .1(100)]} + \underset{\text{(U11 \& U12)}}{2(17)}$$
$$= +54 \longrightarrow 54 \text{ nsec after } t_0$$

The chip 'sees' a data setup time, t_{DW}, of:

(EQ D)
$$t_{DW} = \underset{(t_{WRval})}{128} - \underset{(t_{DATAval})}{54} = 74 \text{ nsec}$$

This satisfies the 50-nsec minimum data setup time requirement of the RAM chip. However, if the delay due to transceiver enable time is considered, we find that valid data does not arrive at the RAM chip until:

(EQ E)
$$t_{DATAval} = \underset{\text{(data available on S-100 bus)}}{[+30 - .1(100)]} + \underset{\text{(U10)}}{33} + \underset{\text{(U7)}}{25} + \underset{\text{(U11 output enable delay)}}{45} + \underset{\text{(U12)}}{17}$$
$$= +140 \longrightarrow 140 \text{ nsec after } t_0$$

In this case, the RAM chip does not see valid data until after the write strobe has passed! By changing U10 to a 74F-series device (74F14) and U6/U11/U12 to 74AS-series devices (74AS245), the 'valid data available' time becomes:

(EQ F)
$$t_{DATAval} = \underset{\text{(data available on S-100 bus)}}{[+30 - .1(100)]} + \underset{\text{(U10)}}{7.5} + \underset{\text{(U7)}}{25} + \underset{\text{(U11 output enable delay)}}{13} + \underset{\text{(U12)}}{11}$$
$$= +76.5 \longrightarrow 76.5 \text{ nsec after } t_0$$

In this case, the chip sees a data setup time, t_{DW}, of 51.5 nsec which is greater than the 50-nsec minimum requirement of the RAM chip. Actually, we have a little more margin over the 'minimum data setup' time requirement than the above result indicates. This is because we assumed **minimum** U10 propagation delay for the t_{WRval} calculation and **maximum** U10 propagation delay for the $t_{DATAval}$ calculation.

Based on information from the 74AS245 data sheets, the data bus lines draw 750 uA when driven low. This exceeds the S-100 bus loading specification of 500 uA. The loading problem is solved by substituting 74ALS parts for U6/U11 and a 74F part for U12. Overall AC timing will remain within specification, while S-100 line DC loading is reduced to 100 uA.

Using a 74F14 for U10 causes the DC loading on the pWR* line to go out of specification. Changing U10 to a 74AS04 reduces bus loading to 100 uA while maintaining the same AC timing as the 74F14 part. We lose the benefit of the Schmitt trigger (increased noise immunity), but this is acceptable in view of the reduced bus loading that results.

To verify that the chip's t_{AW} (address setup time requirement) is met, we find that valid address information is not available to the RAM chips until:

(EQ G)
$$t_{ADDRval} = \underset{\text{(S-100 bus addr valid)}}{-70} + \underset{\text{(U10/11)}}{33}$$
$$= -37 \longrightarrow 37 \text{ nsec before } t_0$$

Thus:

(EQ H)
$$t_{AW} = \underset{(t_{WRval})}{+128} - \underset{(t_{ADDRval})}{(-37)} = 165 \text{ nsec}$$

This meets the 85-nsec t_{AW} requirement of the RAM chip.

The write pulse width seen by the RAM chip is:

(EQ I)
$$t_{WP} = \underset{\text{(pWR* pulse width)}}{.9(100)} - \underset{\text{(U10)*}}{(15 - 4)} - \underset{\text{(U8)*}}{(7.5 - 2.5)}$$
$$= 74 \text{ nsec}$$

*Difference between the gate's min and max propagation delay.

This satisfies the chip's t_{WP} requirement of 70 nsec.

be considered when verifying compliance of the design with S-100 bus timing requirements. Since gate propagation delay depends on the TTL logic family used, the easiest approach is to assume a logic family and then make adjustments as necessary.

A timing analysis is presented on the text box of page 50. The discussion assumes a bus clock speed of 10 MHz (t_{CY}=100 nsec), worst-case figures for all IEEE-696 bus timing specifications, and worst-case propagation delay for 74LS-family TTL gates. I also assume that the 120-nsec version of the RAM chip is used (a 100-nsec version is available, but it costs about twice as much as the 120-nsec part). For simplicity, only the analysis of an 8-bit odd-addressed write cycle is shown, but in practice all possible bus cycle types must be analyzed. For the discussion, t_0 is defined as the high-to-low transition of pSTVAL* when pSYNC is high (Figure 1). All S-100 bus signal times are referenced to t_0. Refer to Figures 1, 9, and 10 as you read through the timing analysis on page 50.

As shown, the design meets all write-cycle timing requirements at a bus speed of 10 MHz. However, after performing a similar analysis for an 8-bit, even-addressed (worst case) memory read bus cycle, we would find that the delay to t_{CSval} (Equation A) is too long to permit use of 120-nsec RAM chips. To avoid the need for faster RAM chips, U1, U3, U4, and U8 are upgraded to 74F-series logic, and U2 is upgraded to 74AS series.

The 74F283 (U1) causes the DC loading on address lines A16-A19 to be out of specification under worst-case conditions. In fact, we find that the DC loading specification cannot be met without significantly degrading the speed of the circuit. However, I consider this violation minor because few boards on the bus use these lines; I/O boards, for example, use only A0-A15 since the I/O address bus is only 16 bits wide. If you require closer compliance with the DC loading S-100 bus specification on these lines, the 74F283 may be replaced with a 74LS283. However, DC loading will still be about 300 uA out of specification and use of 100 nsec RAM chips may be necessary to ensure satisfactory operation at 10 MHz.

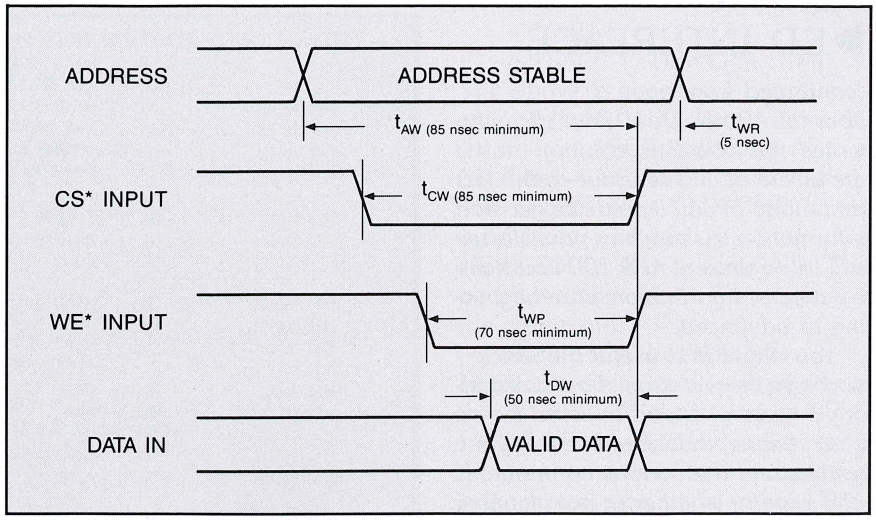


Figure 10. *Write-cycle timing requirements of the NEC 120-nsec RAM chip.*

Finally, we must consider the power requirements of the entire circuit since only one voltage regulator will fit on the board. Fortunately, each RAM chip draws only 100 uA when not selected. When selected, each can draw as much as 70 mA. Adding up all the power requirements (absolute maximum) for all chips results in a maximum current requirement of 768 mA. Thus, a single 7805, 5-Volt/1-Amp regulator will satisfy all power requirements.

All that remains at this point is to program the PAL, prototype the design, and construct a printed circuit board. (A preprogrammed PAL and printed circuit board are available for this project. Please see Table 1, on page 40, for a list of parts used in the final design of this RAM card, and for information on how to order the PAL and printed circuit board.)

IN CONCLUSION

Although a static RAM design was presented, the same 8/16-bit interface could be used for a 16-bit I/O port. Only minor modification of the bus status decoding portion of the circuit would be required.

Due to ambiguity of the bus standard, the designer must pay particular attention to minimizing sXTRQ* response time. Future revisions of the bus standard should resolve the timing conflicts between the sXTRQ*, SIXTN*, and PHANTOM* lines.

S-100 JOURNAL ARTICLE REPRINTS ARE AVAILABLE

If you find yourself in need of a number of copies of a specific article from S-100 Journal, we can provide reprints in quantity. The reprints are on the same type of paper as the magazine and are reproduced with the same colors. For more information and prices, please write or call.

S-100 Journal
PO Box 1914
Orem, Utah 84057
(801) 373-0696

▶ED INTERFACE

(continued from page 7)
Journal. Both Don Pannell (who writes the 696-Bus column) and I are aware of and sensitive to the fact that many of you would like get-start information. As much as possible, we will try to present in S-100 Journal a balance of information from beginning to advanced.

You might find useful the article I wrote on the first issue about assembling a system. I tried to write it in a basic manner that would help anyone understand the various components of a system, whether or not planning to build one. •*Jay*

Dual-Processor Operation

Thank you and thank Howard Spindel for the article on bringing up CP/M-86 on CompuPro's 8085/8088 CPU card (S-100 Journal, Vol. 1, No. 2). An entire new processor is now available to me with a simple one-line command.

I have found one drawback, however. In transferring between CP/M-2.2 and CP/M-86, it sometimes gets confusing which operating system is active at any given time. To solve this problem, I have modified the prompt character in CP/M-86 to be lowercase drive letters instead of the uppercase letters used in CP/M-2.2. It is now very easy to tell which processor (operating system) I am executing on.

A single byte in the CP/M-86 operating system object file is modified for the patch. CP/M-2.2's DDT.COM program is used as follows:

- Customize the CBIOS86.A86 file as needed for your system (may not be necessary — see the article).
- Follow the article, generating all the files including CPMX.CMD (the customized object file for CP/M-86).
- Under CP/M-2.2, execute DDT.COM with the following commands:

```
A>DDT CPMX.CMD
DDT VERS 2.2
NEXT PC
2D80 0100
—S4E9
04E9 41 61
04EA E8 .
```

▶

— LISTING CONTINUED FROM PAGE 28 —

```
016D 7B        EPILOG: MOV     A,E
016E 320A05            STA     STPCNT
0171 7A                MOV     A,D
0172 320B05            STA     STPCNT+1        ;STORE STEP COUNT


0175 0E09              MVI     C,PRINTF        ;NOSTRP MESSAGE
0177 11AB04            LXI     D,NOSTRP
017A CD0500            CALL    BDOS

017D 2A0A05            LHLD    STPCNT          ;PRINT SPTCNT IN HEX
0180 CDB903            CALL    HEXOUT          ;PRINT HEX VALUE
0183 CDA803            CALL    CRLF            :CARRIAGE RETURN, LINE FEED

0186 3A0A05            LDA     STPCNT          ;FETCH LOWER 8 BITS OF STPCNT
0189 47                MOV     B,A             ;STORE AS BYTE COUNT

018A 210E05            LXI     H,BUFFER
018D CDE401            CALL    DECODE          :DISPLAY BARS

0190 0E09              MVI     C,PRINTF        ;GOTIT MESSAGE
0192 118F04            LXI     D,GOTIT         ;MESSAGE BUFFER
0195 CD0500            CALL    BDOS

0198 0E01              MVI     C,CONINP        ;READ KEYBRD
019A CD0500            CALL    BDOS
019D CDA803            CALL    CRLF

01A0 FE59              CPI     'Y'             ;EXIT?
01A2 C21F01            JNZ     PROLOG
01A5 CDA803            CALL    CRLF

01A8 2A0E06    FINIS:  LHLD    OLDSP           ;RESTORE STACK
01AB F9                SPHL
01AC C9                RET                     ;RETURN TO DOS

               ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
               ;       BLACK STRIP ROUTINE:  CUSTOMIZE?                                 ;
               ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
01AD 010000    BSTRIP: LXI     B,0             ;CLEAR TIMER
01B0 DBA4      BLOOP:  IN      WAND            ;READ WAND
01B2 03                INX     B
01B3 FEFF              CPI     BLACK
01B5 C2C101            JNZ     BOUT            ;IF NOT BLACK, EXIT
01B8 3A1000            LDA     10H             ;16*256 LOOPS?
01BB B8                CMP     B
01BC C2B001            JNZ     BLOOP           ;IF NOT KEEP CHECKING
01BF 37                STC                     ;SET CARRY TO INDICATE TIMEOUT
01C0 C9                RET
01C1 CDDF01    BOUT:   CALL    PACK

01C4 B7                ORA     A               ;CLEAR CARRY TO INDICATE STRIPE CHANGE
01C5 C9                RET

               ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
               ;       WHITE STRIP ROUTINE: CUSTOMIZE?                                  ;
               ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
01C6 010000    WSTRIP: LXI     B,0             ;CLEAR TIMER
01C9 DBA4      WLOOP:  IN      WAND            ;READ WAND
01CB 03                INX     B
01CC FEFE              CPI     WHITE
01CE C2DA01            JNZ     WOUT            ;IF NOT WHITE, EXIT
01D1 3A1000            LDA     10H             ;16*256 LOOPS?
01D4 B8                CMP     B
01D5 C2C901            JNZ     WLOOP           ;IF NOT KEEP CHECKING
01D8 37                STC                     ;SET CARRY TO INDICATE TIMEOUT
01D9 C9                RET
01DA CDDF01    WOUT:   CALL    PACK
01DD B7                ORA     A               :CLEAR CARRY TO INDICATE STRIPE CHANGE
01DE C9                RET

01DF 71        PACK:   MOV     M,C             ;STORE HIGH BYTE
01E0 23                INX     H

01E1 70                MOV     M,B             ;STORE LOW BYTE
01E2 23                INX     H
01E3 C9                RET

               DECODE:                         ;HL= BUFFER B=# OF HEX NO.
01E4 3E3A              MVI     A,58            ;MINIMUM NUMBER OF BARS TO SCAN
01E6 B8                CMP     B
```

```
01E7 D2FA01          JNC     DERR1          ;ERROR MESSAGE IF NOT ENOUGH BARS
01EA CD0302          CALL    GUARD3         ;3 GUARD BARS
01ED CD2302          CALL    DIG6L          ;SIX LEFT DIGITS
01F0 CD1002          CALL    GUARD5         ;5 GUARD BARS
01F3 CD3F02          CALL    DIG6R          ;SIX RIGHT DIGITS
01F6 CD0302          CALL    GUARD3         ;3 GUARD BARS
01F9 C9              RET
01FA 0E09     DERR1: MVI     C,PRINTF       ;GET READY TO PRINT ERROR MESSAGE
01FC 11BE04          LXI     D,ERR1
01FF CD0500          CALL    BDOS
0202 C9              RET

              GUARD3:
0203 CD9903          CALL    SPRINT         ;PRINT HEX ENTITY
0206 CD9903          CALL    SPRINT         ;PRINT HEX ENTITY

0209 CD9903          CALL    SPRINT         ;PRINT HEX ENTITY
020C CDA803          CALL    CRLF           ;CARRIAGE RETURN, LINEFEED
020F C9              RET

              GUARD5:
0210 CD9903          CALL    SPRINT
0213 CD9903          CALL    SPRINT
0216 CD9903          CALL    SPRINT
0219 CD9903          CALL    SPRINT
021C CD9903          CALL    SPRINT
021F CDA803          CALL    CRLF
0222 C9              RET

0223 0E06     DIG6L: MVI     C,6            ;PRINT SIX DIGIT LINES
0225 220405   DIGLL: SHLD    HLREG
0228 CD9903          CALL    SPRINT
022B CD9903          CALL    SPRINT
022E CD9903          CALL    SPRINT
0231 CD9903          CALL    SPRINT
0234 CD5B02          CALL    CODE
0237 CDA803          CALL    CRLF
023A 0D              DCR     C

023B C22502          JNZ     DIGLL
023E C9              RET

023F 0E06     DIG6R: MVI     C,6            ;PRINT SIX DIGIT LINES
0241 220405   DIGRL: SHLD    HLREG
0244 CD9903          CALL    SPRINT
0247 CD9903          CALL    SPRINT
024A CD9903          CALL    SPRINT
024D CD9903          CALL    SPRINT
0250 CD5B02          CALL    CODE
0253 CDA803          CALL    CRLF
0256 0D              DCR     C
0257 C24102          JNZ     DIGRL
025A C9              RET

025B E5       CODE:  PUSH    H              ;SAVE HL REG
025C C5              PUSH    B
025D 2A0405          LHLD    HLREG          ;FETCH OLD HLREG
0260 0604            MVI     B,4            ;BYTE COUNTER
0262 5E       PSTACK: MOV    E,M            ;FETCH LOW BYTE
0263 23              INX     H              ;MOVE POINTER
0264 56              MOV     D,M            ;FETCH HIGH BYTE

0265 23              INX     H              ;MOVE POINTER
0266 D5              PUSH    D              ;PUSH ITEM TO BE ADDED ON STACK
0267 05              DCR     B              ;DECREMENT BYTE COUNTER
0268 C26202          JNZ     PSTACK         ;LOOP
026B 210000          LXI     H,0            ;CLEAR HL REG
026E 0604            MVI     B,4            ;SET UP BYTE COUNTER
0270 D1       PADD:  POP     D              ;FETCH ADD ITEM
0271 19              DAD     D              ;HL=HL+DE
0272 05              DCR     B              ;DECREMENT BYTE COUNTER
0273 C27002          JNZ     PADD           ;LOOP
0276 220205          SHLD    HLSUM          ;STORE SUM
0279 CD6403          CALL    DIV8Y7
027C C5              PUSH    B              ;       <---------------+
027D C5              PUSH    B              ;       <-------------+ !
027E E1              POP     H              ;HL=BC  -------------+ !
027F 220605          SHLD    BCREG          ;                    !
0282 D1              POP     D              ;DE=BC  -------------+
0283 B7              ORA     A              ;CLEAR CARRY
0284 7A              MOV     A,D            ;SHIFT MS BIT
0285 1F              RAR
```

—CTRL C

A>SAVE 45 NEWCPMX.CMD

The above sequence of commands changes byte 04E9 (hex) from a 41 (hex — an ASCII uppercase A) to a 61 (hex — an ASCII lowercase a). The original file CPMX.CMD is not modified, instead a new file NEWCPMX.CMD is created. The new operating system is executed by the command:

BOOT86 NEWCPMX.CMD

Howard had also stated a requirement that an EPROM containing an 8088 FAR JUMP to address 0 is necessary if more than 64K of RAM is installed in the system (see article section: 'Providing for an 8088 Reset or Restart Vector'). I tried booting CP/M-86 with more than 64K of RAM without installing this EPROM. It worked! Then I tried to figure out why the EPROM was unnecessary. Most S-100 systems have a terminated bus. This means that reading data from a nonexistent memory location results in the return of FF (hexadecimal). The 8088 starts out reading address FFFF0 which is where the EPROM is supposed to go. Without the EPROM, the 8088 reads the undefined op-code FFFF and proceeds to do this 8 times, doing nothing until it loops around in memory back to address 0. Now the 8088 has succeeded in reaching address 0 without the need of the EPROM. I suggest this approach first, prior to spending the time connecting a special EPROM into the system. NOTE: This trick works with the CompuPro CPU 8085/88 card. I do not know if it will work with the Macrotech MI-286 card.

I hope this helps some of your readers.

Don Pannell
San Jose, California

bits

The Bits *department is for publishing* ***non-commercial*** *small advertisements. There is* ***no charge*** *for subscribers to place an ad. However, please limit your message to* ***50 words.*** *If your ad has more words, there is a charge of $1.00 for each word over 50.*

You may take advantage of this service to sell or buy used S-100 hardware, trade personal programs, etc. Send ads and prepayment (if applicable) to S-100 Journal, BITS, PO Box 1914, Orem, UT 84057.

FOR SALE

Cromemco 68000/Z80 multiuser system. Cromix/plus operating system, w/ C, Pascal, and BASIC. Runs CP/M applications. Contact Robert Coats, 919-544-2443.

33 Megabyte hard disk. 14 inch PRIAM 3350, in cabinet with ADES PS100 S-100 controller, works great, cost $5000, sell for $1100. ICM CPZ48000 Z80 single board computer in Integrand 800DB2F cabinet with two Siemens FDD100-8 drives, works good, cost $1500, sell $750. Roland Bjorklund, Indian Head, Maryland. 301-753-9461.

S-100 TurboDOS 3-user system; CCS 2200 series 12-slot (mini)mainframe, Systems Group Z80 CPU, 65k RAM, 5/8-inch DD floppy, Teletek HD/CTC (hard disk/QIC-02 tape) controller, 3 Earth Turbo (Z80H/8MHz 128K) Slave boards, 2 QUME DT-8 DS-DD (1.2Mb) floppies, 1 or 2 — 20Mb hard disks, TurboDOS v 1.41 O/S, RTC, and lots of s/w. Price: 20Mb system is $2,500.00, 40Mb is $2,750.00. Steve Sanders (813) 791-1938.

S-100 CP/M system, includes: Computime 4MHz Z80 board, DRC 64K static RAM board, Ackerman ►

```
0286 67          MOV    H,A          ;STORE TOP HALF OF HL
0287 7B          MOV    A,E          ;SHIFT LS BIT

0288 1F          RAR
0289 6F          MOV    L,A          ;STORE BOT HALF OF HL
028A 220805      SHLD   HDBY7        ;SAVE
028D 2A0405      LHLD   HLREG
0290 CDE103      CALL   SPACE        ;PRINT A SPACE
0293 3E00        MVI    A,0
0295 320C05      STA    DIGIT        ;CLEAR DIGIT
0298 CDE103      CALL   SPACE
029B CD7303      CALL   COMPUTE      ;CONVERT TIMING TO UNIT GUARD LENGTH
029E E5          PUSH   H
029F D5          PUSH   D
02A0 E1          POP    H
02A1 CDB903      CALL   HEXOUT       ;PRINT GUARD LENGTH
02A4 E1          POP    H
02A5 7B          MOV    A,E          ;FETCH 1ST BAR WIDTH
02A6 320C05      STA    DIGIT
02A9 CDE103      CALL   SPACE
02AC CD7303      CALL   COMPUTE      ;CONVERT TIMING TO UNIT GUARD LENGTH
02AF E5          PUSH   H
02B0 D5          PUSH   D
02B1 E1          POP    H
02B2 CDB903      CALL   HEXOUT       ;PRINT GUARD LENGTH

02B5 E1          POP    H
02B6 7B          MOV    A,E          ;FETCH 2ND BAR WIDTH
02B7 E603        ANI    03H          ;MASK
02B9 0707        RLC ! RLC
02B8 5F          MOV    E,A
02BC 3A0C05      LDA    DIGIT        ;PACK BITS IN DIGIT
02BF B3          ORA    E            ;VARIABLE
02C0 320C05      STA    DIGIT
02C3 CDE103      CALL   SPACE
02C6 CD7303      CALL   COMPUTE      ;CONVERT TIMING TO UNIT GUARD LENTH
02C9 E5          PUSH   H
02CA D5          PUSH   D
02CB E1          POP    H
02CC CDB903      CALL   HEXOUT       ;PRINT GUARD LENGTH
02CF E1          POP    H
02D0 7B          MOV    A,E          ;FETCH 3RD BAR WIDTH
02D1 E603        ANI    03H          ;MASK
02D3 07070707    RLC ! RLC ! RLC! RLC
02D7 5F          MOV    E,A
02D8 3A0C05      LDA    DIGIT        ;PACK BITS IN DIGIT
02DB B3          ORA    E            ;VARIABLE
02DC 320C05      STA    DIGIT

02DF CDE103      CALL   SPACE
02E2 CD7303      CALL   COMPUTE      ;CONVERT TIMINT TO UNIT GUARD LENGTH
02E5 E5          PUSH   H
02E6 D5          PUSH   D
02E7 E1          POP    H
02E8 CDB903      CALL   HEXOUT       ;PRINT GUARD LENGTH
02EB E1          POP    H
02EC 7B          MOV    A,E          ;FETCH 4TH BAR WIDTH
02ED E603        ANI    03H          ;MASK
02EF 0707070707  RLC ! RLC ! RLC! RLC ! RLC ! RLC
02F5 5F          MOV    E,A
02F6 3A0C05      LDA    DIGIT        ;PACK BITS IN DIGIT
02F9 B3          ORA    E            ;VARIABLE
02FA 320C05      STA    DIGIT
02FD 2A0C05      LHLD   DIGIT
0300 CDE103      CALL   SPACE
0303 CDE103      CALL   SPACE
0306 CDB903      CALL   HEXOUT       ;PRINT CONTENTS OF DIGIT


0309 0E09        MVI    C,PRINTF
030B 11F904      LXI    D,VALUE

030E CD0500      CALL   BDOS

0311 3A0C05      LDA    DIGIT        ;IS IT A ZERO?
0314 1E30        MVI    E,'0'
0316 FE5B        CPI    05BH
0318 CA5C03      JZ     PVALUE

031B 1E31        MVI    E,'1'        ;IS IT AN ONE?
031D FE6A        CPI    06AH
031F CA5C03      JZ     PVALUE
```

```
0322 1E32            MVI    E,'2'          ;IS IT A TWO?
0324 FEA6            CPI    0A6H
0326 CA5C03          JZ     PVALUE

0329 1E33            MVI    E,'3'          ;IS IT A THREE?
032B FE5D            CPI    05DH
032D CA5C03          JZ     PVALUE

0330 1E34            MVI    E,'4'          ;IS IT A FOUR?
0332 FEB5            CPI    0B5H
0334 CA5C03          JZ     PVALUE


0337 1E35            MVI    E,'5'          ;IS IT A FIVE?
0339 FE79            CPI    079H
033B CA5C03          JZ     PVALUE

033E 1E36            MVI    E,'6'          ;IS IT A SIX?
0340 FED5            CPI    0D5H
0342 CA5C03          JZ     PVALUE

0345 1E37            MVI    E,'7'          ;IS IT A SEVEN?
0347 FE9D            CPI    09DH
0349 CA5C03          JZ     PVALUE

034C 1E38            MVI    E,'8'          ;IS IT AN EIGHT?
034E FED9            CPI    0D9H
0350 CA5C03          JZ     PVALUE

0353 1E39            MVI    E,'9'          ;IS IT A NINE?
0355 FE97            CPI    097H
0357 CA5C03          JZ     PVALUE

035A 1E3F            MVI    E,'?'          ;MUST BE UNDEFINED

035C 0E02    PVALUE: MVI    C,CONOUT
035E CD0500          CALL   BDOS

0361 C1              POP    B
0362 E1              POP    H
0363 C9              RET

0364 11F9FF  DIVBY7: LXI    D,-7           ;SUBTRACT 7
0367 010000          LXI    B,0
036A 19      DIVLP:  DAD    D              ;HL=HL+(-7)
036B 7C              MOV    A,H            ;TEST HIGH BYTE
036C 03              INX    B
036D E680            ANI    80H            ;NEGATIVE?
036F CA6A03          JZ     DIVLP
0372 C9              RET

             COMPUTE:                      ;HL=BUFFER, BC=DIVISOR, HDBY7= .5 BC
0373 5E              MOV    E,M            ;FETCH LSB
0374 23              INX    H              ;MOVE POINTER
0375 56              MOV    D,M            ;FETCH MSB
0376 23              INX    H              ;MOVE POINTER
0377 E5              PUSH   H              ;SAVE HL

0378 2A0805          LHLD   HDBY7          ;ADD .5 BC (ROUND OFF)
037B 19              DAD    D              ;BAR + .5 BC
037C 11FFFF          LXI    D,0FFFFH

037F 3A0705          LDA    BCREG+1        ;NEGATE BC
0382 2F              CMA
0383 47              MOV    B,A
0384 3A0605          LDA    BCREG
0387 2F              CMA
0388 4F              MOV    C,A

0389 09      CLOOP:  DAD    B              ;SUBTRACT BC
038A 7C              MOV    A,H            ;FETCH MS BYTE
038B 13              INX    D
038C E680            ANI    80H            ;TEST SIGN
038E CA8903          JZ     CLOOP
0391 E1              POP    H
0392 7B              MOV    A,E            ;TEST IF REG E IS 4
0393 FE04            CPI    4
0395 C0              RNZ                   ;IF NOT IGNORE
0396 1E03            MVI    E,3            ;SET REG E TO 4 -- WHITE SPACE APPEARS LARGER
                                                    ;AND OFTEN A 3 SEEMS LIKE A 4
```

Promblaster II, 2 Shugart 801 disk drives, 1 Megabyte RAM-disk, Viewpoint terminal. Scott Baker, 18185 West Union Rd, Portland, OR 97229 (503) 645-0734.

Cromemco Z2—extra heavy duty S-100 enclosure with 21 slot motherboard and 30 Amp power supply. Cards mount vertically from the front for easy access; unusual like new condition $395.00. D. A. Danello, Rt 1 Box 217B, Blacksburg, VA 24060.

S-100 System — Teletek Systemaster in a compact 4-slot cabinet with dual 8-inch half-height 1.2 Meg floppies. Comes with both CP/M and TurboDOS operating system. $500 to a good home. 716-593-5329 evenings.

22 slot IMSAI box with front panel, Z80 CPU card, CompuPro Interfacer 3 8-port serial card, all cables & documentation. All in excellent condition. Asking $750. Must sell — best offer. Call or write: Michael C. Greenspon, 2124 Kittredge #117, Berkeley, CA 94704, (415) 849-2182.

Teletek Systemaster SBC 64K RAM, Integrand enclosure and power supply, 3-8" DS/DD Tandon TM848-2D disc drives, IMS 481 board, 4-port serial I/O, CSD graphics board set (NEC7220, up to 1024×1024 pixels), CP/M, misc. software, NCR industrial dot matrix printer. All for $600/offer. Paul Smith 612-479-1523.

WANTED

The CP/M-2.2 software I received with my Sierra Data Sciences SBC-100 is not compatible with my ROM BIOS. Can anyone provide me with a software and/or ROM upgrade so I can modify my BIOS? Glenn Mitchell, Maine Medical Center, Portland, ME 04102. ►

Hardware and Software for the Marinchip M9900CPU board. Specifically interested in 64K RAM cards, any software (especially languages). All must be functional. Call or write Chris Bobbitt, 9120 Rhode Island Ave., College Park, MD 20740, (301) 345-2492 evenings.

Wanted: PLINK or LYNX overlay linkers for CP/M-80. Will trade or buy — contact D. A. Danello, Rt 1 Box 217-B, Blacksburg, VA 24060.

Wanted: CPZ-48000 IMS CPU board, 4MHz. Sell/trade: Teletek Systemaster I CPU board, X comp SA1000/ST506 hard disk interface cards, Shugart 1004 hard disk. Travel Systems, 818-502-3090.

I would appreciate if anyone would share with me their experience in the implementation of the S-100 Bus in the legal environment. I am particularly interested in word processing and any good time, billing and accounting package that may be in use by other attorneys. George E. Mangan, Attorney at Law, 47 North 200 East, Roosevelt, UT 84066, (801) 722-2428.

HELP. I recently purchased an older S-100 machine with no documentation. It was built by Computer Data Systems of Addison, Texas, and roughly resembles a TRS-80 model 3. The CPU was built by CDS, floppy drives & controller by Micropolis. It is supposed to run CP/M or MDOS. I would like to obtain a bootable copy of CP/M. R. L. Gallahan, Box 1701, Rockville, MD 20850.

```
0398 C9              RET

0399 5E       SPRINT: MOV     E,M             ;FETCH LO BYTE
039A 23               INX     H
039B 56               MOV     D,M             ;FETCH HI BYTE
039C 23               INX     H
039D EB               XCHG
039E CDB903           CALL    HEXOUT          ;PRINT A NIBBLE
03A1 EB               XCHG
03A2 3E20             MVI     A,' '
03A4 CDD203           CALL    CHOUT           ;PRINT A SPACE
03A7 C9               RET

03A8 F5C5D5E5 CRLF:   PUSH PSW ! PUSH B ! PUSH D ! PUSH H
03AC 0E09             MVI     C,PRINTF        ;CRLF
03AE 11B804           LXI     D,CRLFM
03B1 CD0500           CALL    BDOS
03B4 E1D1C1F1         POP H   ! POP D  ! POP B  ! POP PSW
03B8 C9               RET

03B9 7C       HEXOUT: MOV     A,H
03BA CDBE03           CALL    PHEX            ;PRINT LEFT BYTE

03BD 7D               MOV     A,L
03BE F5       PHEX:   PUSH    A               ;SAVE REG A
03BF 0F0F0F0F         RRC ! RRC ! RRC ! RRC   ;GET LEFT NIBBLE
03C3 CDC703           CALL    PNIB            ;PRINT NIBBLE
03C6 F1               POP     A               ;RESTORE REG A
03C7 E60F     PNIB:   ANI     0FH             ;MASK LOWER FOUR BITS
03C9 C630             ADI     30H             ;CONVERT TO INTEGER
03CB FE3A             CPI     '9'+1           ;TEST UPPER LIMIT OF INTEGER
03CD FAD203           JM      CHOUT
03D0 C607             ADI     7               ;CONVERT TO A-F
03D2 F5C5D5E5 CHOUT:  PUSH PSW ! PUSH B ! PUSH D ! PUSH H
03D6 5F               MOV     E,A
03D7 0E02             MVI     C,CONOUT
03D9 CD0500           CALL    BDOS
03DC E1D1C1F1         POP H   ! POP D  ! POP B  ! POP PSW
03E0 C9               RET

03E1 F5       SPACE:  PUSH    PSW             ;SAVE REGISTER A
03E2 3E20             MVI     A,' '           ;CHARACTER TO PRINT
03E4 CDD203           CALL    CHOUT           ;ACTUAL CALL
03E7 F1               POP     PSW             ;RESTORE REG A
03E8 C9               RET


03E9 57454C434FSIGNON DB      'WELCOME TO THE BAR-CODE WAND PROGRAM',0DH,0AH,0AH
0410 2020504C41       DB      '  PLACE WAND OVER WHITE AREA, AND PRESS WAND BUTTON',0DH,0AH
0445 2020574149       DB      '  WAIT FOR READY MESSAGE',0DH,0AH,0AH,'$'

0461 2020535449DARK   DB      '  STILL DARK',0DH,'$'

046F 0D0A0A2020READY  DB      0DH,0AH,0AH,'  READY!',07H,'  START SCANNING ',0DH,0AH,'$'

048F 2020454E44GOTIT  DB      '  END OF SCAN-- QUIT (Y/N)?$'

04AB 0D0A0A2320NOSTRP DB      0DH,0AH,0AH,'# OF BARS = $'

04BB 0D0A24   CRLFM   DB      0DH,0AH,'$'

04BE 0D0A2A2A2AERR1   DB      0DH,0AH,'*** ERROR-- DID NOT SCAN ENOUGH BARS',0DH,0AH,'$'

04E7 2020544F54SUM    DB      '  TOTAL = $'

04F2 202F37203DDIV7   DB      ' /7 = $'

04F9 2044494749VALUE  DB      ' DIGIT= $'


0502          HLSUM   DS      2
0504          HLREG   DS      2
0506          BCREG   DS      2
0508          HDBY7   DS      2
050A          STPCNT  DS      2
050C          DIGIT   DS      2
050E          BUFFER  DS      256
060E          OLDSP   DS      2       ;OLD STACK POINTER (FOR RETURN TO CPM)
0610                  DS      64      ;STACK AREA
              STKTOP                  ;CURRENT STACK LOCATION
0650                  END
```

BARCODE MAGIC FOR S-100

LISTING 2

CWAND — THIS IS THE BARCODE WAND DRIVER. IT IS DESIGNED TO BE PATCHED INTO CP/M AND BECOME PART OF THE OPERATING SYSTEM.

```
                ;       CWAND   -- CPM WAND PATCH TO MAKE THE BAR CODE
                ;               WAND PART OF CPM OPERATING SYSTEM
FFFF =          TRUE    EQU     0FFFFH
0000 =          FALSE   EQU     00000H
0000 =          DEBUG   EQU     FALSE
0001 =          CONIN:  EQU     1
0002 =          CONOT:  EQU     2
007D =          CSTAT:  EQU     07DH
007C =          CDATA:  EQU     07CH
0001 =          TXMASK: EQU     1H
0002 =          RXMASK: EQU     2H
0005 =          BDOS:   EQU     0005H
D90A =          CONINP: EQU     0D90AH
D90C =          CONOUT: EQU     0D90CH
00A4 =          WAND:   EQU     0A4H
00FE =          WHITE:  EQU     0FEH
00FF =          BLACK:  EQU     0FFH
                ;
                ;       CPM PATCHES
                ;

                ;       ORG     0D906H
                ;       JMP     WCONIN  ;INTERCEPT ORIGINAL CONSOLE IN
                ;
                ;       THE REAL STUFF
                ;
                ;       ORG     0F800H  ;FREE RAM
                ;
0100                    ORG     0100H
                ;       DUMMY PROGRAM
0100 311B05             LXI     SP,STKTOP
0103 211101             LXI     H,WCONIN
0106 220AD9             SHLD    CONINP

0109 0E01       LOOP:   MVI     C,CONIN
010B CD0500             CALL    BDOS
010E C30901             JMP     LOOP

                ;START OF REAL STUFF

0111 E5D5C5     WCONIN: PUSH H! PUSH D! PUSH B!

0114 3A2A04             LDA     BUFFLG

0117 FE00               CPI     0       ;BUFFER EMPTY
0119 C2A403             JNZ     PROCES  ;PROCESS A CHARACTER
011C DBA4       CIN:    IN      WAND    ;CHECK WAND STATUS
011E FEFE               CPI     WHITE
0120 CC3A01             CZ      WANDD   ;IF BLACK PROCESS
0123 3A2A04             LDA     BUFFLG
0126 FE00               CPI     0
0128 C2A403             JNZ     PROCES
012B DB7D               IN      CSTAT   ;CHECK STATUS
012D E602               ANI     RXMASK  ;KEY PRESSED?
012F CA1C01             JZ      CIN     ;IF NOT KEEP CHECKING
0132 DB7C               IN      CDATA   ;INPUT CHARACTER
0134 E67F               ANI     07FH    ;MASK PARITY
0136 C1D1E1     EXIT:   POP B! POP D! POP H!
0139 C9                 RET

013A CDA201     WANDD:  CALL    CLEAR
013D 210000             LXI     H,0
0140 23         SYNC:   INX     H
0141 3E10               MVI     A,10H
0143 BC                 CMP     H
0144 CA5401             JZ      WAIT

0147 DBA4               IN      WAND
0149 FEFF               CPI     BLACK
014B C8                 RZ
014C CA4001             JZ      SYNC
014F 0E07               MVI     C,07
0151 CD0CD9             CALL    CONOUT
0154 DB7D       WAIT:   IN      CSTAT
0156 E602               ANI     RXMASK
0158 C0                 RNZ
0159 DBA4               IN      WAND
015B FEFF               CPI     BLACK
015D C25401             JNZ     WAIT

0160 216304     PROLOG: LXI     H,BARS  ;INIT POINTER TO BAR STORAGE AREA
0163 1E01               MVI     E,1     ;CLEAR STRIPE TIMER
0165 37         BODY:   STC             ;SET CARRY BIT
0166 CDC201             CALL    BSTRIP  ;BLACK STRIPE TIMER
0169 DA8401             JC      EPILOG  ;IF CARRY BIT IS SET THEN TIMEOUT
016C 1C                 INR     E       ;INCREMENT STRIP COUNTER
016D 7B                 MOV     A,E     ;TEST COUNTER
016E FE3B               CPI     59      ;ENOUGH STRIPES?
0170 D28401             JNC     EPILOG

0173 37                 STC             ;SET CARRY BIT
0174 CDDB01             CALL    WSTRIP  ;WHITE STRIPE TIMER
0177 DA8401             JC      EPILOG  ;IF CARRY BIT IS SET THEN TIMEOUT
017A 1C                 INR     E       ;INCREMENT STRIP COUNTER
017B 7B                 MOV     A,E     ;TEST COUNTER
017C FE3B               CPI     59      ;ENOUGH STIPES?
017E D28401             JNC     EPILOG
0181 C36501             JMP     BODY

0184 7B         EPILOG: MOV     A,E     ;SAVE
0185 324404             STA     STPCNT  ;STPCNT

0188 47                 MOV     B,A     ;FETCH STRIPE COUNT
0189 216304             LXI     H,BARS
018C CDF901             CALL    DECODE
018F CDA201             CALL    CLEAR
0192 0E07               MVI     C,07
0194 CD0CD9             CALL    CONOUT
0197 DBA4       WAITT:  IN      WAND
0199 FEFF               CPI     BLACK
019B C29701             JNZ     WAITT
019E CDE003             CALL    CHKBUF

01A1 C9                 RET

01A2 210000     CLEAR:  LXI     H,0
01A5 224504             SHLD    WCOUNT
01A8 214904             LXI     H,RBUF
01AB 224704             SHLD    RBUFF
01AE 216204             LXI     H,LBUF
01B1 225504             SHLD    LBUFF
01B4 216304             LXI     H,BARS  ;SET BEGINNING OF CLEAR BUFFER
01B7 3E00               MVI     A,0     ;CLEAR CHARACTER
01B9 0678               MVI     B,120   ;NUMBER OF LOCATIONS
01BB 77         CLRLP:  MOV     M,A     ;CLEAR OPERATION
01BC 23                 INX     H       ;MOVE POINTER
01BD 05                 DCR     B       ;DECREMENT COUNTER
01BE C2BB01             JNZ     CLRLP   ;LOOP
01C1 C9                 RET

01C2 010000     BSTRIP: LXI     B,0     ;CLEAR TIMER
01C5 DBA4       BLOOP:  IN      WAND    ;READ WAND
01C7 03                 INX     B
01C8 FEFF               CPI     BLACK
01CA C2D601             JNZ     BOUT    ;IF NOT BLACK, EXIT

01CD 3A1000             LDA     10H     ;16*256 LOOPS ?
01D0 B8                 CMP     B
01D1 C2C501             JNZ     BLOOP   ;IF NOT KEEP CHECKING
01D4 37                 STC             ;SET CARRY, INDICATING CONT
01D5 C9                 RET
01D6 CDF401     BOUT:   CALL    PACK    ;STORE TIMER COUNT
01D9 B7                 ORA     A       ;CLEAR TO INDICATE CHANGE
01DA C9                 RET
```

```
01DB 010000    WSTRIP: LXI     B,0     ;CLEAR TIMER
01DE DBA4      WLOOP:  IN      WAND    ;READ WAND
01E0 03                INX     B
01E1 FEFE              CPI     WHITE
01E3 C2EF01            JNZ     WOUT    ;IF NOT WHITE, EXIT
01E6 3A1000            LDA     10H     ;16*256 LOOPS ?
01E9 B8                CMP     B
01EA C2DE01            JNZ     WLOOP   ;IF NOT KEEP CHECKING
01ED 37                STC             ;SET CARRY, INDICATING CONT
01EE C9                RET
01EF CDF401    WOUT:   CALL    PACK    ;STORE TIMER COUNT
01F2 B7                ORA     A       ;CLEAR TO INDICATE CHANGE
01F3 C9                RET


01F4 71        PACK:   MOV     M,C     ;STORE HIGH BYTE
01F5 23                INX     H
01F6 70                MOV     M,B     ;STORE LOW BYTE
01F7 23                INX     H
01F8 C9                RET

               DECODE:                 ;HL=BUFFER B=# OF HEX NO.
01F9 3E3A              MVI     A,58    ;MINIMUM NUMBER OF BARS TO SCAN
01FB B8                CMP     B
01FC D0                RNC             ;RETURN OF NOT ENOUGH BARS
01FD CD1302            CALL    GUARD3  ;3 GUARD BARS
0200 CD1D02            CALL    DIG6L   ;6 LEFT DIGITS
0203 CD0D02            CALL    GUARD5  ;5 GUARD BARS
0206 CD3602            CALL    DIG6R   ;6 RIGHT DIGITS
0209 CD1302            CALL    GUARD3  ;3 GUARD BARS
020C C9                RET

               GUARD5:
020D CD9B03            CALL    SPRINT
0210 CD9B03            CALL    SPRINT
0213 CD9B03    GUARD3: CALL    SPRINT

0216 CD9B03            CALL    SPRINT
0219 CD9B03            CALL    SPRINT
                       IF      DEBUG
                       CALL    CRLF
                       ENDIF
021C C9                RET

021D 0E06      DIG6L:  MVI     C,6
021F 223B04    DIGLL:  SHLD    HLREG
0222 CD9B03            CALL    SPRINT
0225 CD9B03            CALL    SPRINT
0228 CD9B03            CALL    SPRINT
022B CD9B03            CALL    SPRINT
022E CD4F02            CALL    CODE
                       IF      DEBUG
                       CALL    CRLF
                       ENDIF
0231 0D                DCR     C
0232 C21F02            JNZ     DIGLL
0235 C9                RET

0236 0E06      DIG6R:  MVI     C,6

0238 223B04    DIGRL:  SHLD    HLREG
023B CD9B03            CALL    SPRINT
023E CD9B03            CALL    SPRINT
0241 CD9B03            CALL    SPRINT
0244 CD9B03            CALL    SPRINT
0247 CD4F02            CALL    CODE
                       IF      DEBUG
                       CALL    CRLF
                       ENDIF
024A 0D                DCR     C
024B C23802            JNZ     DIGRL
024E C9                RET

024F E5        CODE:   PUSH    H       ;SAVE HL REG
0250 C5                PUSH    B
```

```
0251 2A3B04            LHLD    HLREG   ;FETCH OLD HLREG
0254 0604              MVI     B,4     ;BYTE COUNTER
0256 5E        PSTACK  MOV     E,M     ;FETCH LOW BYTE
0257 23                INX     H       ;MOVE POINTER
0258 56                MOV     D,M     ;FETCH HIGHT BYTE
0259 23                INX     H       ;MOVE POINTER
025A D5                PUSH    D       ;PUSH ITEM TO BE ADDED ON STACK

025B 05                DCR     B       ;DECREMENT BYTE COUNTER
025C C25602            JNZ     PSTACK  ;LOOP
025F 210000            LXI     H,0     ;CLEAR HL REG
0262 0604              MVI     B,4     ;ADD UP FOUR BARS
0264 D1        PADD:   POP     D       ;FETCH ADD ITEM
0265 19                DAD     D       ;ADD
0266 05                DCR     B       ;DECREMENT BYTE COUNTER
0267 C26402            JNZ     PADD    ;LOOP
026A 223D04            SHLD    HLSUM   ;STORE SUM
026D CD6603            CALL    DIVBY7
0270 C5                PUSH    B
0271 C5                PUSH    B
0272 E1                POP     H       ;HL<=BC
0273 223F04            SHLD    BCREG
0276 D1                POP     D
0277 B7                ORA     A       ;CLEAR CARRY
0278 7A                MOV     A,D     ;SHIFT MS BIT
0279 1F                RAR
027A 67                MOV     H,A
027B 7B                MOV     A,E
027C 1F                RAR
027D 6F                MOV     L,A     ;STORE BOT HALF OF HL

027E 224204            SHLD    HDBY7   ;SAVE
0281 2A3B04            LHLD    HLREG
0284 3E00              MVI     A,0
0286 324104            STA     DIGIT   ;CLEAR DIGIT
0289 CD7503            CALL    COMPUTE
028C 7B                MOV     A,E
028D 324104            STA     DIGIT   ;STORE 1ST BAR WIDTH
0290 CD7503            CALL    COMPUTE
0293 7B                MOV     A,E     ;FETCH 2ND BAR WIDTH
0294 E603              ANI     03H     ;MASK
0296 0707              RLC! RLC !      ;SHIFT
0298 5F                MOV     E,A
0299 3A4104            LDA     DIGIT   ;FETCH DIGIT
029C B3                ORA     E       ;ADD 2ND BAR WIDTH
029D 324104            STA     DIGIT
02A0 CD7503            CALL    COMPUTE
02A3 7B                MOV     A,E     ;FETCH 3RD BAR WIDTH
02A4 E603              ANI     03H     ;MASK
02A6 07070707          RLC ! RLC ! RLC ! RLC
02AA 5F                MOV     E,A
02AB 3A4104            LDA     DIGIT
02AE B3                ORA     E

02AF 324104            STA     DIGIT
02B2 CD7503            CALL    COMPUTE
02B5 7B                MOV     A,E     ;FETCH 4TH BAR WIDTH
02B6 E603              ANI     03H     ;MASK
02B8 0707070707        RLC ! RLC ! RLC ! RLC ! RLC
02BE 5F                MOV     E,A
02BF 3A4104            LDA     DIGIT
02C2 B3                ORA     E
02C3 324104            STA     DIGIT

02C6 1E30              MVI     E,'0'
02C8 FE5B              CPI     05BH
02CA CA0E03            JZ      PVALUE

02CD 1E31              MVI     E,'1'
02CF FE6A              CPI     06AH
02D1 CA0E03            JZ      PVALUE

02D4 1E32              MVI     E,'2'
02D6 FEA6              CPI     0A6H
02D8 CA0E03            JZ      PVALUE
```

```
02DB 1E33           MVI     E,'3'
02DD FE5D           CPI     05DH
02DF CA0E03         JZ      PVALUE

02E2 1E34           MVI     E,'4'
02E4 FE85           CPI     085H
02E6 CA0E03         JZ      PVALUE

02E9 1E35           MVI     E,'5'
02EB FE79           CPI     079H
02ED CA0E03         JZ      PVALUE

02F0 1E36           MVI     E,'6'
02F2 FED5           CPI     0D5H
02F4 CA0E03         JZ      PVALUE

02F7 1E37           MVI     E,'7'
02F9 FE9D           CPI     09DH
02FB CA0E03         JZ      PVALUE

02FE 1E38           MVI     E,'8'
0300 FED9           CPI     0D9H

0302 CA0E03         JZ      PVALUE

0305 1E39           MVI     E,'9'
0307 FE97           CPI     097H
0309 CA0E03         JZ      PVALUE

030C 1E3F           MVI     E,'?'
030E 4B       PVALUE: MOV   C,E
030F CDCA03         CALL    RPACK

                            ;RIGHT TO LEFT SCAN
0312 0E20           MVI     C,' '
0314 3A4104         LDA     DIGIT

0317 1E30           MVI     E,'0'
0319 FEE5           CPI     0E5H

031B CA5F03         JZ      PVAL1

031E 1E31           MVI     E,'1'
0320 FEA9           CPI     0A9H
0322 CA5F03         JZ      PVAL1

0325 1E32           MVI     E,'2'
0327 FE9A           CPI     09AH
0329 CA5F03         JZ      PVAL1

032C 1E33           MVI     E,'3'
032E FE75           CPI     075H
0330 CA5F03         JZ      PVAL1

0333 1E34           MVI     E,'4'
0335 FE5E           CPI     05EH
0337 CA5F03         JZ      PVAL1

033A 1E35           MVI     E,'5'
033C FE6D           CPI     06DH
033E CA5F03         JZ      PVAL1


0341 1E36           MVI     E,'6'
0343 FE57           CPI     057H
0345 CA5F03         JZ      PVAL1

0348 1E37           MVI     E,'7'
034A FE76           CPI     076H
034C CA5F03         JZ      PVAL1

034F 1E38           MVI     E,'8'
0351 FE67           CPI     067H
0353 CA5F03         JZ      PVAL1
```

```
0356 1E39           MVI     E,'9'
0358 FED6           CPI     0D6H
035A CA5F03         JZ      PVAL1

035D 1E3F           MVI     E,'?'

035F 4B       PVAL1 MOV     C,E
0360 CDD503         CALL    LPACK

0363 C1             POP     B
0364 E1             POP     H
0365 C9             RET

0366 11F9FF   DIVBY7: LXI   D,-7    ;SUBTRACT 7
0369 010000         LXI     B,0
036C 19       DIVLP:  DAD   D       ;HL=HL+(-7)
036D 7C             MOV     A,H     ;TEST HIGH BYTE
036E 03             INX     B
036F E680           ANI     80H     ;NEGATIVE?
0371 CA6C03         JZ      DIVLP
0374 C9             RET

                    COMPUTE:        ;HL=BUFFER, BC=DIVISOR, HDBY7=
0375 5E             MOV     E,M     ;FETCH LSB
0376 23             INX     H       ;MOVE POINTER
0377 56             MOV     D,M     ;FETCH MSB
0378 23             INX     H       ;MOVE POINTER
0379 E5             PUSH    H       ;SAVE HL
037A 2A4204         LHLD    HDBY7   ;ADD .5 BC (ROUND OFF)

037D 19             DAD     D       ;BAR + .5 BC
037E 11FFFF         LXI     D,0FFFFH

0381 3A4004         LDA     BCREG+1 ;NEGATE BC
0384 2F             CMA
0385 47             MOV     B,A
0386 3A3F04         LDA     BCREG
0389 2F             CMA
038A 4F             MOV     C,A

038B 09       CLOOP:  DAD   B       ;SUBTRACT BC
038C 7C             MOV     A,H     ;FETCH MS BYTE
038D 13             INX     D
038E E680           ANI     80H     ;TEST SIGN
0390 CA8B03         JZ      CLOOP
0393 E1             POP     H
0394 7B             MOV     A,E     ;TEST IF REG E IS 4
0395 FE04           CPI     4
0397 C0             RNZ             ;IF NOT IGNORE
0398 1E03           MVI     E,3     ;SET REG E TO 4-- WHITE SPACE
                                     ALWAYS SEEM LARGER
                                    ;AND OFTEN A 3 SEEMS LIKE A 4
039A C9             RET


039B 5E       SPRINT: MOV   E,M     ;FETCH LO BYTE
039C 23             INX     H
039D 56             MOV     D,M     ;FETCH HI BYTE
039E 23             INX     H
039F EB             XCHG
03A0 EB             XCHG
03A1 3E20           MVI     A,' '
03A3 C9             RET

03A4 2A2B04   PROCES: LHLD  BUFLOC  ;FETCH BUFFER LOCATION
03A7 7E             MOV     A,M
03A8 322904         STA     CHAR    ;SAVE IN TEMPORARY LOCATION
03AB 23             INX     H       ;INCREMENT BUFFER LOCATION
03AC 222B04         SHLD    BUFLOC  ;UPDATE BUFFER LOCATION VARIABLE


03AF 113B04         LXI     D,BUFEND ;FETCH BUFFER END LOCATION
03B2 7A             MOV     A,D
03B3 BC             CMP     H       ;D=H?
03B4 C2C403         JNZ     PROEXIT
03B7 7B             MOV     A,E
```

```
03B8 BD                 CMP    L        ;E=L?
03B9 C2C403             JNZ    PROEXIT
03BC 3E00               MVI    A,0      ;CLEAR BUFFER FLAG
03BE 322A04             STA    BUFFLG
03C1 C31C01             JMP    CIN

03C4 3A2904     PROEXIT LDA    CHAR
03C7 C33601             JMP    EXIT

03CA E5         RPACK   PUSH   H
03CB 2A4704             LHLD   RBUFF
03CE 73                 MOV    M,E
03CF 23                 INX    H
03D0 224704             SHLD   RBUFF
03D3 E1                 POP    H
03D4 C9                 RET


03D5 E5         LPACK   PUSH   H
03D6 2A5504             LHLD   LBUFF
03D9 73                 MOV    M,E
03DA 2B                 DCX    H
03DB 225504             SHLD   LBUFF
03DE E1                 POP    H
03DF C9                 RET

03E0 214904     CHKBUF: LXI    H,RBUF   ;CHECK FOR '?'
03E3 060C               MVI    B,12
03E5 7E         CHKL1:  MOV    A,M      ;FETCH CHARACTER
03E6 FE3F               CPI    '?'
03E8 CAFC03             JZ     CHKBF2   ;IF SO JUMP
03EB 23                 INX    H        ;MOVE POINTER
03EC 05                 DCR    B        ;DECREMENT POINTER
03ED C2E503             JNZ    CHKL1    ;LOOP
03F0 060C       CHK1:   MVI    B,12     ;IF YOU GET HERE, BUFFER OK
03F2 212D04             LXI    H,BUFFER         ;FETCH DESTINATION ADDRESS
03F5 EB                 XCHG            ; DE <=> HL
03F6 214904             LXI    H,RBUF   ;SOURCE ADDRESS
03F9 C31304             JMP    CHKL
03FC 215704     CHKBF2: LXI    H,LLBUF  ;CHECK LEFT BUFFER FOR '?'

03FF 060C               MVI    B,12
0401 7E         CHKL2:  MOV    A,M      ;FETCH CHARACTER
0402 FE3F               CPI    '?'
0404 C8                 RZ              ;IF ? GET OUT
0405 23                 INX    H        ;MOVE POINTER
0406 05                 DCR    B        ;DECREMENT POINTER
0407 C20104             JNZ    CHKL2    ;LOOP
040A 060C       CHK2:   MVI    B,12
040C 212D04             LXI    H,BUFFER         ;FETCH DESTINATION ADDRESS
040F EB                 XCHG            ; DE <=> HL
0410 215704             LXI    H,LLBUF  ;SOURCE ADDRESS

0413 7E         CHKL:   MOV    A,M      ;FETCH SOURCE BYTE
0414 23                 INX    H        ;MOVE POINTER
0415 EB                 XCHG            ; DE <=> HL
0416 77                 MOV    M,A      ;STORE DESTINATION
0417 23                 INX    H        ;MOVE POINTER
0418 EB                 XCHG            ; DE <=> HL
0419 05                 DCR    B        ;DECREMENT COUNTER
041A C21304             JNZ    CHKL
041D 3E01               MVI    A,1
041F 322A04             STA    BUFFLG

0422 212D04             LXI    H,BUFFER
0425 222B04             SHLD   BUFLOC
0428 C9                 RET

0429            CHAR    DS     1
042A 00         BUFFLG  DB     0        ;0=EMPTY BUFFER
042B 2D04       BUFLOC  DW     BUFFER
042D 3031323334BUFFER   DB     '012345678987',0DH
043A                    DS     1
043B =          BUFEND  EQU    $
043B            HLREG   DS     2
043D            HLSUM   DS     2
043F            BCREG   DS     2
0441            DIGIT   DS     1
0442            HDBY7   DS     2
0444            STPCNT  DS     1
0445            WCOUNT  DS     2
0447            RBUFF   DS     2
0449            RBUF    DS     12
0455            LBUFF   DS     2
0457            LLBUF   DS     11
0462            LBUF    DS     1
0463            BARS    DS     120
04DB                    DS     64
051B            STKTOP  DS     2
051D                    END
```

BARCODE MAGIC FOR S-100

LISTING 3

PATCH — THIS IS A PROGRAM TO INSTALL THE BARCODE WAND DRIVER TEMPORARILY INTO CP/M.

```
;       PATCH-- CPM PROGRAM TO INSTALL CWAND.COM INTO CPM
;               CWAND.COM IS ATTACHED TO THIS .COM VIA DDT
;               AT 200H
;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;                                                                        ;
;       MAKE SURE THE CONINP, BDOS, AND WCONIN ADDRESSES ARE CORRECT     ;
;       THE NUMBERS HERE ARE FOR THE SD SYSTEMS  SBC 200  CPU OPERATING  ;
;       UNDER A 56K CP/M SYSTEM                                          ;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

D90A =          CONINP  EQU    0D90AH  ;ORIGINAL CONSOLE INPUT ROUTINE IN BIOS
BE00 =          BDOS    EQU    0BE00H  ;WHERE C WAND LOADS IN.  THE FIRST STMT
                                       ;IS A JUMP TO THE REAL BIOS
BE03 =          WCONIN  EQU    0BE03H  ;NEW CONSOLE  INPUT ROUTINE THAT ALSO
                                       ;MONITORS  WANDS.
0100                    ORG    0100H

0100 210000             LXI    H,0     ;SAVE OLD STACK
0103 39                 DAD    SP

0104 223301             SHLD   OLDSP
0107 314901             LXI    SP,STKTOP;LOAD IN NEW STACK

010A 2103BE             LXI    H,WCONIN ;FIX BIOS JUMP ENTRY
010D 220AD9             SHLD   CONINP

0110 2100BE             LXI    H,BDOS  ;FETCH BDOS CALL ALSO DESTINATION
0113 220600             SHLD   0006H   ;BDOS PATCH

0116 EB                 XCHG
0117 210002             LXI    H,0200H   ;SOURCE

011A 010005             LXI    B,0500H ;NUMBER OF BYTES

011D 7E         LOOP:   MOV    A,M     ;FETCH SOURCE
011E 23                 INX    H
011F EB                 XCHG
0120 77                 MOV    M,A     ;STORE DEST
0121 23                 INX    H
0122 EB                 XCHG
0123 0B                 DCX    B       ;CHECK IF TIME TO QUIT
0124 3E00               MVI    A,0

0126 B8                 CMP    B
0127 C21D01             JNZ    LOOP
012A B9                 CMP    C
012B C21D01             JNZ    LOOP

012E 2A3301             LHLD   OLDSP
0131 F9                 SPHL
0132 C9                 RET

0133            OLDSP   DS     2
0135                    DS     20
0149            STKTOP  DS     2
014B                    END
```

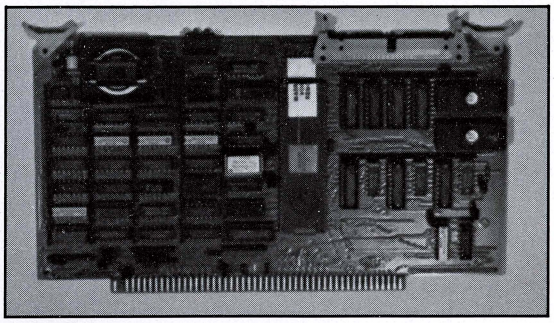
BARCODE MAGIC FOR S-100
LISTING 4

BARCODE.BAS — THIS PROGRAM PRINTS BARCODES ON THE IDS 460G PRINTER. REQUIRES ADAPTATION FOR OTHER PRINTERS.

```
        REM     THIS PROGRAM PRINTS BARCODES ON THE LINE PRINTER
        REM     JUST ENTER THE NUMBER AND THE PROGRAM DOES THE
        REM     REST.

        REM     500 --GRAPHICS ON ROUTINE
        REM     600 --GRAPHICS OFF ROUTINE
        REM     700 --OUTPUT CHAR TO LINE PRINTER
        REM     800 --WHITE BAR
        REM     900 --BLACK BAR
        REM     1000--DIGIT (LEFT SIDE)
        REM     2000--DIGIT (RIGHT SIDE)

        DIM DIGIT(12)

100     INPUT "ENTER BARCODE NUMBER (12 DIGITS)"; NUM$
        IF LEN(NUM$) > 12 THEN 100
        IF LEN(NUM$) = 0 THEN STOP

200     FOR I%=1 TO 12
          DIGIT(I%) = VAL(MID$(NUM$,I%,1))
          IF (DIGIT(I%) < 0 OR DIGIT(I%) > 9) THEN GOTO 100
        NEXT I%


        INPUT "ENTER DESCRIPTION "; DESC$

        PRINT "NUMBER =";NUM$
        PRINT "DESCRIPTION =";DESC$

        INPUT "ENTRY CORRECT (Y/N)?"; ANS$
        IF ANS$ <> "Y" THEN 100

        LPRINTER WIDTH 0
        PRINT USING "&" ;CHR$(31),DESC$,CHR$(30)
300     FOR J%=1 TO 6
          FOR K%= 1 TO 3
            GOSUB 900   REM BLACK BAR
            GOSUB 800   REM WHITE BAR
            GOSUB 900   REM BLACK BAR

310         FOR I%=1 TO 6
              GOSUB 1000
            NEXT I%

400         GOSUB 800   REM WHITE BAR
            GOSUB 900   REM BLACK BAR

            GOSUB 800   REM WHITE BAR
            GOSUB 900   REM BLACK BAR
            GOSUB 800   REM WHITE BAR

410         FOR I%=7 TO 12
              GOSUB 2000
            NEXT I%

450         GOSUB 900   REM BLACK BAR
            GOSUB 800   REM WHITE BAR
            GOSUB 900   REM BLACK BAR
            PRINT USING "&&"; CHR$(3),CHR$(13),
          NEXT K%
          PRINT USING "&&"; CHR$(3),CHR$(14),
        NEXT J%
        GOSUB 600       REM GRAPHICS OFF
        PRINT
        PRINT USING "&"; CHR$(1),MID$(NUM$,1,6),"  ",MID$(NUM$,7,6),CHR$(2),
        PRINT
        PRINT
        PRINT
        CONSOLE
        GOTO 100

        REM     GRAPHICS ON
500     PRINT USING "&"; CHR$(3),
        RETURN

        REM     GRAPHICS OFF
600     PRINT USING "&"; CHR$(3),CHR$(2),
        RETURN

        REM     CHARACTER OUT
700     PRINT USING "&&"; CHR$(CHAR%),CHR$(CHAR%),
        RETURN

        REM     BLACK BAR
800     CHAR%=0
        GOSUB 700
        RETURN

        REM     WHITE BAR
900     CHAR%=127

        GOSUB 700
        RETURN

        REM     LEFT SIDE
1000    ON DIGIT(I%)+1 GOSUB 1100,1110,1120,1130,1140,1150,
                             1160,1170,1180,1190
        RETURN

1100    GOSUB 800
        GOSUB 800
        GOSUB 800
        GOSUB 900
        GOSUB 900
        GOSUB 800
        GOSUB 900
        RETURN

1110    GOSUB 800
        GOSUB 800
        GOSUB 900
        GOSUB 900
        GOSUB 800
        GOSUB 800

        GOSUB 900
        RETURN

1120    GOSUB 800
        GOSUB 800
        GOSUB 900
        GOSUB 800
        GOSUB 800
        GOSUB 900
        GOSUB 900
        RETURN

1130    GOSUB 800
        GOSUB 900
        GOSUB 900
        GOSUB 900
        GOSUB 900
        GOSUB 800
        GOSUB 900
        RETURN

1140    GOSUB 800

        GOSUB 900
        GOSUB 800
        GOSUB 800
        GOSUB 800
        GOSUB 900
        GOSUB 900
        RETURN
```

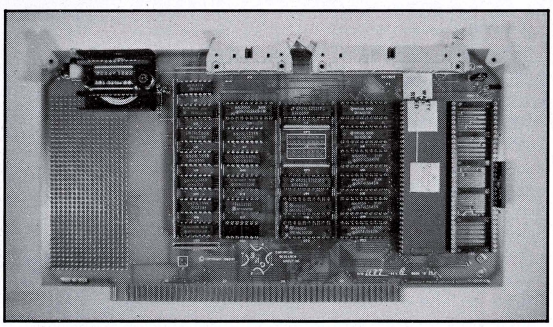
IEEE-696 S-100 BOARDS

The most efficient 68000 Card Set implementation on the market



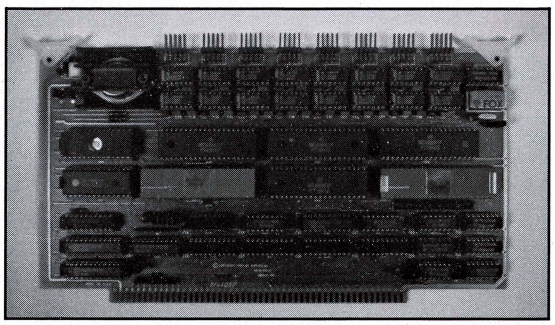
★ **ERG68-696 CPU**

68000/68010 MPU
8, 10, 12 MHz clock
ergFORTH ROM kernel and compiler
'SKY-BUS' for system expansion
Full DMA bus transfer timing logic
Auto adjusts to speed of device on bus
User-definable delay generators Price: $785.00



★ **MMU (Memory Management Unit)**

UNIX with optimized 68451
Full user memory protection
Dynamic memory allocation Price: $625.00



★ **ACC8 Intelligent I/O Channel Card**

68008 Coprocessor (8 or 10 MHz)
Dual ported RAM
8 RS-232 serial ports
Double buffered parallel interface
On-board RAM firmware
DUART (68681) Price: $730.00

OEM PRICING ON REQUEST

NU COM CORPORATION
19944 N.E. Ballinger Way • Seattle, Washington 98155
206-364-9911


```
1150    GOSUB 800
        GOSUB 900
        GOSUB 900
        GOSUB 800
        GOSUB 800
        GOSUB 800
        GOSUB 900
        RETURN

1160    GOSUB 800
        GOSUB 900
        GOSUB 800
        GOSUB 900
        GOSUB 900

        GOSUB 900
        GOSUB 900
        RETURN

1170    GOSUB 800
        GOSUB 900
        GOSUB 900
        GOSUB 900
        GOSUB 800
        GOSUB 900
        GOSUB 900
        RETURN

1180    GOSUB 800
        GOSUB 900
        GOSUB 900
        GOSUB 800
        GOSUB 900
        GOSUB 900
        GOSUB 900
        RETURN


1190    GOSUB 800
        GOSUB 800
        GOSUB 800
        GOSUB 900
        GOSUB 800
        GOSUB 900
        GOSUB 900
        RETURN

2000    ON DIGIT(I%)+1 GOSUB 2100,2110,
                       2120,2130,2140,
                       2150,2160,2170,
                       2180,2190
        RETURN

2100    GOSUB 900
        GOSUB 900
        GOSUB 900
        GOSUB 800
        GOSUB 800
        GOSUB 900
        GOSUB 800
        RETURN

2110    GOSUB 900

        GOSUB 900
        GOSUB 800
        GOSUB 800
        GOSUB 900
        GOSUB 900
        GOSUB 800
        RETURN

2120    GOSUB 900
        GOSUB 900
        GOSUB 800
        GOSUB 900
        GOSUB 900
        GOSUB 800
        GOSUB 800
        RETURN

2130    GOSUB 900
        GOSUB 800
        GOSUB 800
        GOSUB 800
        GOSUB 800

        GOSUB 900
        GOSUB 800
        RETURN

2140    GOSUB 900
        GOSUB 800
        GOSUB 900
        GOSUB 900
        GOSUB 900
        GOSUB 800
        GOSUB 800
        RETURN

2150    GOSUB 900
        GOSUB 800
        GOSUB 800
        GOSUB 900
        GOSUB 900
        GOSUB 900
        GOSUB 800
        RETURN


2160    GOSUB 900
        GOSUB 800
        GOSUB 900
        GOSUB 800
        GOSUB 800
        GOSUB 800
        GOSUB 800
        RETURN

2170    GOSUB 900
        GOSUB 800
        GOSUB 800
        GOSUB 800
        GOSUB 900
        GOSUB 800
        GOSUB 800
        RETURN

2180    GOSUB 900
        GOSUB 800
        GOSUB 800
        GOSUB 900

        GOSUB 800
        GOSUB 800
        GOSUB 800
        RETURN

2190    GOSUB 900
        GOSUB 900
        GOSUB 900
        GOSUB 800
        GOSUB 900
        GOSUB 800
        GOSUB 800
        RETURN

        END
```

— END OF BARCODE LISTINGS —

32-BIT S-100 EXPANSION

David A. Plomgren

With the introduction of high-speed 32-bit processors, increased performance pressure has been put on the S-100 bus. Through the efforts of many, the bus withstood and in fact prospered during the transition from low-speed 8-bit devices, such as 5-MHz 8085s and 4-MHz Z80s, to higher-speed 16-bit devices such as 10-MHz 68000s and 10-MHz 80286s. The S-100 community has an interest in seeing the S-100 also expand and prosper through the transition to high-speed 32-bit devices such as 20-MHz 80386s and 25-MHz 68020s.

As with the transition between 8 and 16 bits, serious effort must be given to retain compatibility with existing S-100 cards. With an eye to the future, though, the goal must be to obtain the maximum performance from the microprocessor.

In this article, I will present the fixed requirements, solutions, and a possible implementation of S-100 expansion into the 32-bit arena. Finally, some details are presented as a 'straw man' proposal, intended for debate.

Editor's Note:

The 32-Bit Forum *is intended as a discussion towards a standard 32-bit S-100 implementation.*

If you have any good ideas or comments on using 32-bit processors on the S-100 bus, you are invited to either submit articles or send us a letter with your comments or solutions.

We encourage you to use this and previous articles as starting points for the discussion. Please send all materials to:

32-Bit Forum, S-100 Journal
PO Box 1914, Orem, UT 84057

BASIC REQUIREMENTS

To start the discussion, let's look at what the new microprocessors need in a bus, and what the S-100 can realistically do. The S-100 bus clock (Φ) was defined in the IEEE-696/S-100 specification to be no greater than 6-MHz, and each bus cycle must consist of at least three clock cycles. This limits the bus to a bandwidth of 4 Megabytes/second. To improve the bandwidth of the bus, virtually all S-100 CPU designers have increased the bus clock speed past the 6-MHz IEEE-696 limit. Also to increase the bandwidth, some CPU designers have opted for a two-clock-cycle bus cycle (or other methods that allow a transfer every two clock cycles). The Cromemco 68020 XXU achieves its 8.33 Megabytes/second by running the bus clock at 4.17 MHz and allowing two sequential 16-bit transfers every two clock cycles. Macrotech's 8-MHz MI-286 achieves 8 Megabytes/second running the bus clock at 8-MHz and executing bus cycles every two clock cycles. CompuPro's 12-MHz CPU 286 can transfer 12 Megabytes/second the same way.

As a designer of S-100 boards and systems, it is clear to me that, due to timing considerations, 10 to 12 Megabytes/second is the limit the S-100 bus can achieve while retaining compatibility with even recently designed boards. The use of older boards in a system may limit the speed, and thus bandwidth, to even a lesser value. ►

David Plomgren is a Design Engineer at CompuPro. He is the designer of several of CompuPro's CPU boards. David is also an avid water skier.

It is easy to compute the bandwidth required to support the new 32-bit processors (see Table 1). For Motorola's 20-MHz 68020, the required bandwidth is 27 Megabytes/second (the used bandwidth will be less due to the on-board cache, but for optimal performance, a bus must be able to support the peak rate). For Intel's 20-MHz 80386 as well as Motorola's 20-MHz 68030, the required bandwidth is 40 Megabytes/second.

ARCHITECTURAL SOLUTIONS

Architecturally, there are two ways to approach the problem of effectively quadrupling the bandwidth of the S-100 bus.

The first is to provide local memory on the CPU card in the form of a cache that can fully support the maximum bandwidth of the processor. Figure 1 shows this. As long as the hit rate in the cache memory is high enough, the combined cache-miss overhead and low bandwidth to back store (S-100 bus memory) will not significantly degrade the processor performance. Cromemco used this technique on their 68020 XXU (though I cannot comment on their effective hit rate nor cache-miss overhead).

The advantage of this scheme is that neither a new bus specification nor adjunct bus is necessary. This is very desirable since the S-100 community would not have to agree on a new bus specification that could take years to standardize. One disadvantage to this method, however, is that putting a truly effective cache (at least 16K or 32K of 2 or 4-way set-associative, 0-wait-state RAM, with no software intervention on cache misses), in addition to the processor and other functions, in the forty-plus square inches of available S-100 board space is practically impossible given current levels of integration.

The second way to achieve high bandwidth is by providing a bus and memory that can accommodate the full bandwidth of the microprocessor. Figure 2 shows this. Boards that require over-the-top connectors are not new to S-100 (nor to any standard bus-based computer system with restricted board size).

The advantage of an extra bus is that it opens up the 32-bit bus world completely, instead of limiting the 32-bit path to between the cache and microprocessor. This would allow 32-bit disk drive interfaces, graphics interfaces, I/O interfaces, etc. It also offers the highest possible performance for the microprocessor. It gives an address expansion path, as present systems are approaching the 16-Megabyte limit.

Furthermore, the use of a 32-bit bus does not preclude the use of a cache, but it would improve cache performance by allowing 32-bit fetches when a miss occurs. The major disadvantage of an additional physical bus is the effort and time it will take to standardize such a bus.

WHAT IS CACHE MEMORY?

A cache is a small (less than 64K for microcomputers) set of memory that contains a subset of the data stored in main memory. Caches make use of the concept of program locality, that is, programs usually access data and instructions in a small address range multiple times before moving to another address range.

When the microprocessor requests data, the cache controller checks to see if the data is in cache, then provides it quickly if it is (a hit). If data is not in cache (a miss), the cache controller puts the microprocessor in a wait state and fetches the data from main memory, gives it to the microprocessor, and stores it in the cache for the next time the microprocessor needs it. The performance of a cache depends on many parameters.

IMPLEMENTATION

Vaughn (S-100 Journal, Vol. 1 No. 4, Fall 1986) presents one way to implement the extra bus, and I applaud his initiative. His method is to add 16 data lines and 8 address lines between memory and the CPU using an over-the-top ribbon cable, but continue to use the remainder of the S-100 protocol. The 16/32 protocol would be handled in a way similar to the 8/16 protocol, utilizing two S-100 RFU (Reserved for Future Use) lines.

He suggests, correctly, that the S-100 bus clock cannot be driven past 12-MHz without making obsolete virtually all existing S-100 products. He then infers that, under this scheme, the strobes pDBIN and pWR* can be ignored in favor of boards that 'implement their own strobes.' This will not work because bus masters assert false status and address between bus cycles, and due to the possibility of a Bus-Abort S-100 cycle. Thus, the proposal falls short in one very important way: it only doubles the effective

$$\text{Bandwidth in Megabytes/sec} = \frac{\text{CPU Clock Speed in MHz}}{\text{Minimum No. of clocks per bus cycle}} \times \frac{\text{No. bits per transfer}}{\text{8 bits/byte}}$$

Table 1. *How to calculate the S-100 bus bandwidth. For a 20-MHz 68020, the clock speed is 20 MHz, the number of bits per transfer is 32, and the minimum number of clocks per bus cycle is 3. For a 20-MHz 80386, the number of bits per transfer is 32, and the minimum number of clocks per bus cycle is 2.*

bandwidth up to a maximum of 20 to 24 Megabytes/second, half of what current 32-bit microprocessors need for full performance.

Any new 'over-the-top' bus must be complete enough that it can divorce S-100 rigidity. This means that the new bus must be able to run complete cycles without reducing itself to synchronization with the S-100 bus clock, pSYNC, and the other S-100 signals. At the minimum, the new bus must have a generalized address strobe (AS*), a generalized data strobe (DS*), and the ability to extend a cycle through wait states (WAIT*). Deciding whether to run a fast cycle to the 32-bit Extended Bus Interface (EBI) or a slow cycle to the S-100 bus is then decoded by address on the CPU card, without having to rely on feedback from slave cards. The S-100 bus clock can then rest at half the processor speed (up to 25-MHz processors) or at one quarter the speed (up to 50-MHz processors).

The 4-Gigabyte address space of the new processors could be divided between the 16-Megabyte space of the S-100 bus and the remainder of the 4-Gigabyte 32-bit space. Some memory cards (or peripherals) may only be accessible to the CPU over the 32-bit bus (single ported), and some may be accessible to TMA devices over the S-100 bus as well as to the CPU over the 32-bit bus (dual ported). Standard S-100 boards would be accessible to both TMA devices and the CPU over the S-100 bus.

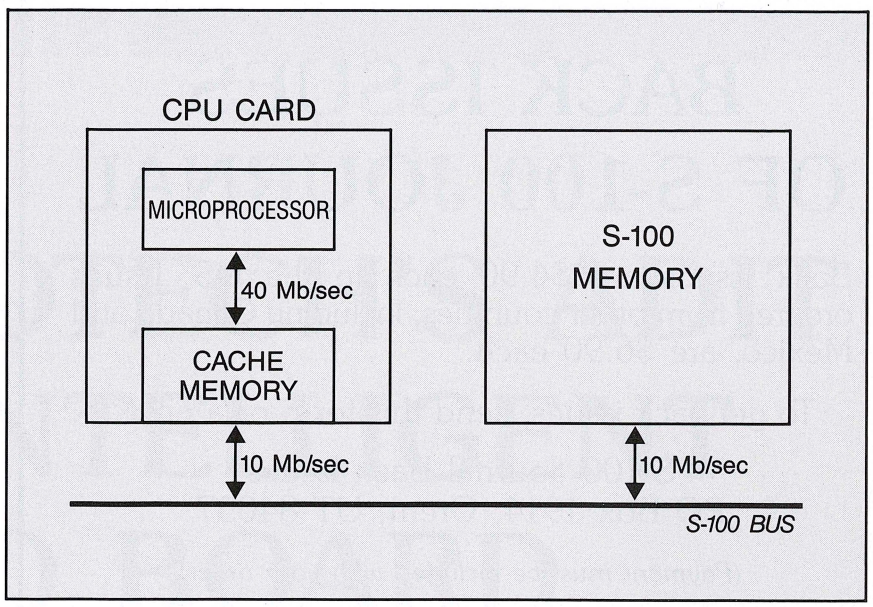


Figure 1. *S-100 bus interface using cache memory.*

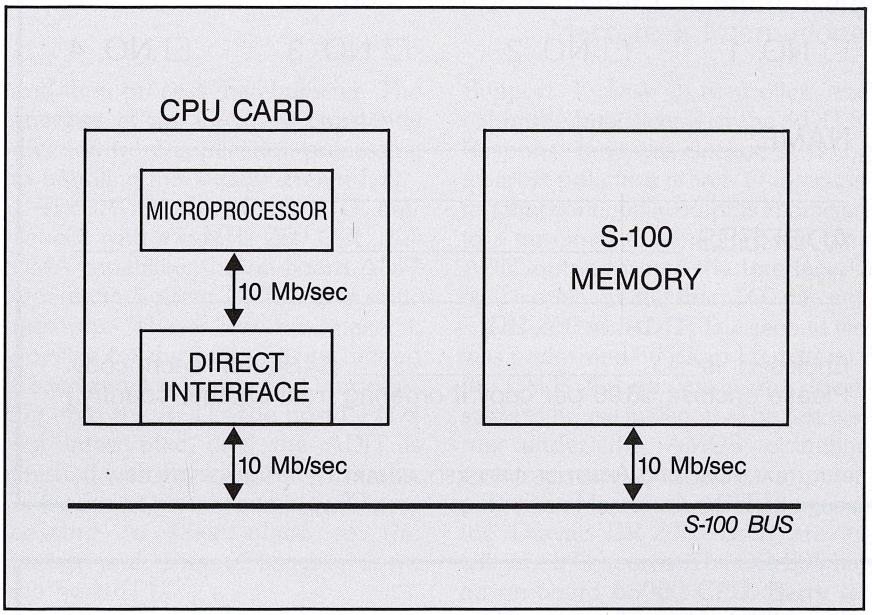


Figure 2. *S-100 bus interface without cache memory.*

DETAILS

Since all current 32-bit processors can transfer 8, 16, 24, and 32 bits of data at a time, I recommend that the EBI also support byte lanes, with four byte-lane status lines. I also recommend that a generalized read/write status be included. Finally, due to noise considerations on the S-100 bus, all 32 bits of data should be brought across the EBI, instead of 16 across the S-100 and 16 across the EBI. Those familiar with the noise induced on the S-100 status (sINP, sOUT, sMEMR, etc.) and strobe (pWR*, pDBIN, etc.) lines by transitions on the data lines (ED0-7, OD0-7) will see the wisdom of this choice.

A 64-pin bus is thus proposed. This includes 32 data lines, 8 address lines, 4 byte lane lines, 1 read/write strobe, 1 address strobe, 1 data strobe, 1 wait line, 13 ground lines, and 3 RFU lines. A standard 64-pin DIN connector with wire-wrap tails then connects two boards together. This method is easily expandable, and has been used successfully on other buses (such as VME). The exact physical configuration and pinout of the EBI should be left to the working group defining the bus.

The timing of this bus is simple and quite technology independent. The CPU guarantees that address and status are valid during the address strobe, and guarantees write-data valid during the data strobe.

For inter-vendor compatibility, CPU and peripheral cards will have an *access time* specification. To secure compatibility, the peripheral will be required to have faster *access time* than the CPU

BACK ISSUES OF S-100 JOURNAL

Back Issues are $4.90 each in the US. Issues ordered from other countries, including Canada and Mexico, are $6.90 each.

To get back issues, send this form or a note to:

S-100 Journal Back Issues
PO Box 1914, Orem, UT 84057

(Payment must be included with your order)

Please send S-100 Journal:

☐ NO. 1 ☐ NO. 2 ☐ NO. 3 ☐ NO. 4

NAME: _____

ADDRESS: _____

Enclosed is _____ ($4.90 for each copy. Please enclose $6.90 per copy if ordering from another country.)

BACK ISSUES CANNOT BE ORDERED AS PART OF A SUBSCRIPTION

Learn to program in DATAFLEX™

The most powerful single- and multi-user database development software

Attend our intensive one and two-day seminars

Call for details and schedules

(415) 495-5999

COCHLIN COMPUTER SYSTEMS

330 TOWNSEND STREET #107 SAN FRANCISCO, CA 94107

demands. All other timings of the bus (such as the WAIT* requirement after status and address are valid) will be described in the specification as a proportion of the *access time*. The *access time* is not literally the read or write time of a peripheral, but rather it is the worst case of a set of proportions relating to the overall speed of the board. Defining the parameters and choosing the appropriate proportions between the parameters would be the task of the working group defining the bus.

S-100 32-BIT STANDARD

Several members of the Microprocessor Standards Committee (MSC) of the Institute of Electrical and Electronic Engineers (IEEE) have expressed an interest in seeing a working group formed to define a 32-bit S-100 standard. Those interested in participating in such a working group, should attend an MSC meeting. (If you are interested, please contact me at 415-786-0909, and I will provide information on how you can become involved.)

CONCLUSION

We, of the S-100 community, must look at what the new 32-bit microprocessors need in the way of a bus, keeping an eye to even higher speed devices. In the spirit of S-100, the bus must be inexpensive to implement, yet provide the full bandwidth required by high-speed 32-bit microprocessors. It also must be compatible with existing S-100 systems. A bus that is independent of S-100 timing is the only solution.

MACROTECH'S ADIT AN INTELLIGENT I/O BOARD

Connie Kelly
Macrotech International

Macrotech International's ADIT is an intelligent I/O board that will support up to 16 terminals, printers, modems, or other serial devices. A ***non***-intelligent I/O board requires the host CPU to spend an inordinate amount of time processing I/O on a character-by-character basis. In small systems with a minimal number of users, this is not a problem. However, as the number of users and peripherals increases, especially in an I/O-intensive situation, the host CPU spends so much time on I/O that response time becomes unacceptably slow. When using nonintelligent I/O, every character to be transferred interrupts the host, requiring the CPU's valuable time to save its current state and then process the character. The energies of the CPU are constantly diverted from application processing to handling this steady stream I/O.

The ADIT solves this I/O bottleneck with a 6-MHz Z80 CPU, full DMA capability, the on-board ADIT Operating System (AOS), and static memory. These features make it possible for the host CPU to offload a substantial amount of I/O processing onto the ADIT. The host CPU is not interrupted until the ADIT is finished with any assigned I/O tasks. This allows time-consuming I/O processing to take place in the background, without bogging down the host CPU.

ABOUT THE TECH FILE

The articles appearing on each issue in Tech File *are produced by companies about one of their own boards. They are not reviews.*

Every S-100 company is invited to submit articles. Please request our 'Tech File Guidelines' before writing an article. Tech File *articles are published one per issue on a first-come-first-served basis.*

BENCHMARKS ILLUSTRATE ADIT SPEED

Before covering the finer points of the ADIT, let's look at some benchmarks that demonstrate the ADIT's talent for increasing system throughput.

Interfacer 3 vs. ADIT. The first test was done in a CompuPro environment running MP/M-86. The system included a Macrotech 80286/Z80H CPU, 1 Megabyte of memory, System Support 1, Disk 1 controller, and either the Interfacer 3 or the ADIT-8. Response time was clocked by doing a barber pole (this is an I/O-intensive test that continually outputs characters to a terminal) to all eight ports. The ADIT outperformed the Interfacer 3 by a margin greater than 160 percent.

DR-200 vs. ADIT. The second test was performed by Barry Hamilton of the TWB Group, an Alpha Micro systems house in Illinois. The test was run under the AMOS operating system on an Alpha Micro computer, pitting the Macrotech ADIT-16 against the Dravac DR-200. Both are intelligent I/O boards. The DR-200 has an on-board 68000 CPU. Barry set up 16 ports all running barber poles at 19.2 Kbaud. Against this I/O-intensive backdrop, a simple-benchmark BASIC program was run. The BASIC program used a FOR-NEXT loop with 25000 iterations:

```
100 FOR I=1 to 25000
110 NEXT
```

The results were decisive. The total execution time for the DR-200 was ►

Connie Kelly is Director of Marketing at Macrotech International. She is also experienced in management and, of course, S-100s. Connie is a flying enthusiast and will soon be getting her pilot's licence.

INTERFACE REGISTERS	
AIC	Adit interface command register.
AIR1,2,3	Adit interface parameter registers.
ASEMA4	Adit semaphore flag register.
AIACK	Adit interrupt acknowledge register.

Table 1. *Interface registers used to pass commands to the ADIT I/O board.*

4 minutes and 42 seconds. The total time for the ADIT was only 34 seconds. As Barry Hamilton stated in his report, 'This is an apparent reduction in system overhead [...] of almost 9/1 in this configuration.

DMA AND ON-BOARD OS OFFLOAD HOST PROCESSING

The ADIT board implements DMA in accordance with IEEE-696 standards, using 24-bit addressing. The I/O channels can transfer data directly to and from memory without tying up the host CPU. When the ADIT board uses DMA, it controls the transaction by issuing strobes, control signals, and data to read or write in system memory. In a DMA transfer, the ADIT first requests the host CPU to relinquish control of the bus. When the host acknowledges the request, the I/O board places the desired address on the bus. When writing into system memory, it also places the data byte on the bus. The board returns control of the bus to the host after the transfer is completed.

Consider an output operation in which 32K bytes of data are sent to a printer. The host CPU would normally be involved in the output as each character was processed, pointing to each character in memory, reading it, and sending it to the printer. However, the ADIT makes this unnecessary. It accepts the output command from the host, and the address of the data. It then directly copies the information from system memory into its own memory while the host continues with other tasks, uninterrupted. After copying the information, the ADIT performs the full output to the printer. The entire output process is transparent to the host, with all interrupts generated by the ADIT's on-board USARTs and associated overhead being handled internally by the ADIT. There are no interrupts on the bus during the transfer. The ADIT interrupts the host only after the full output command has been completed.

ISSUING COMMANDS TO THE ADIT

The ADIT supports two types of commands. The lowest level of data transfer is performed by *immediate* commands. These commands manipulate I/O one character at a time. The majority, however, are *extended* commands. Extended commands use DMA to transfer character strings.

The host initiates I/O processing by issuing an immediate command through the port interface. These commands are defined in the I/O driver. In addition, the extended set of commands recognized by the ADIT's AOS gives the programmer great flexibility in configuring the I/O process for a given system.

Four 8-bit interface registers (Table 1) on the ADIT are used to pass immediate commands and responses between the host and AOS. A 'mailbox' scheme is used: the host deposits a command and parameters in these four registers, the mailbox. The AOS performs the command and outputs information and status back into the registers. The host need not

EXTENDED COMMAND CONTROL BLOCK	
ECR0	Byte count register (low byte).
ECR1	Byte count register (high byte).
ECR2	Source address (low byte).
ECR3	Source address (middle byte).
ECR4	Source address (high byte).
ECR5	Destination address (low byte).
ECR6	Destination address (middle byte).
ECR7	Destination address (high byte).
ECMD	Extended command.
ECRCL, ECRCM	Byte count registers, Count low, Count middle. 0 to 64K. DMA maximum is 64K.
ECRA0,1,2	LMH of data buffer; Address low-middle-high.
ECEOM	Must contain the EOM byte value for those specific commands that require an end-of-message condition test.
ECPARM	Parameter byte-command specific.
ECMASK1	Not yet used. Filled with OFFH.
ECMASK2	Not yet used. Filled with OFFH.
ECW0 - ECW15	Bit definitions of the sixteen 8530 write registers in the Serial Communications Controller Technical Reference Manual.

Table 2. *The Command Block resident in system memory and used to process extended commands. Registers ECR0 through ECR7 are for M-channel (memory-to-memory) data transfers.*

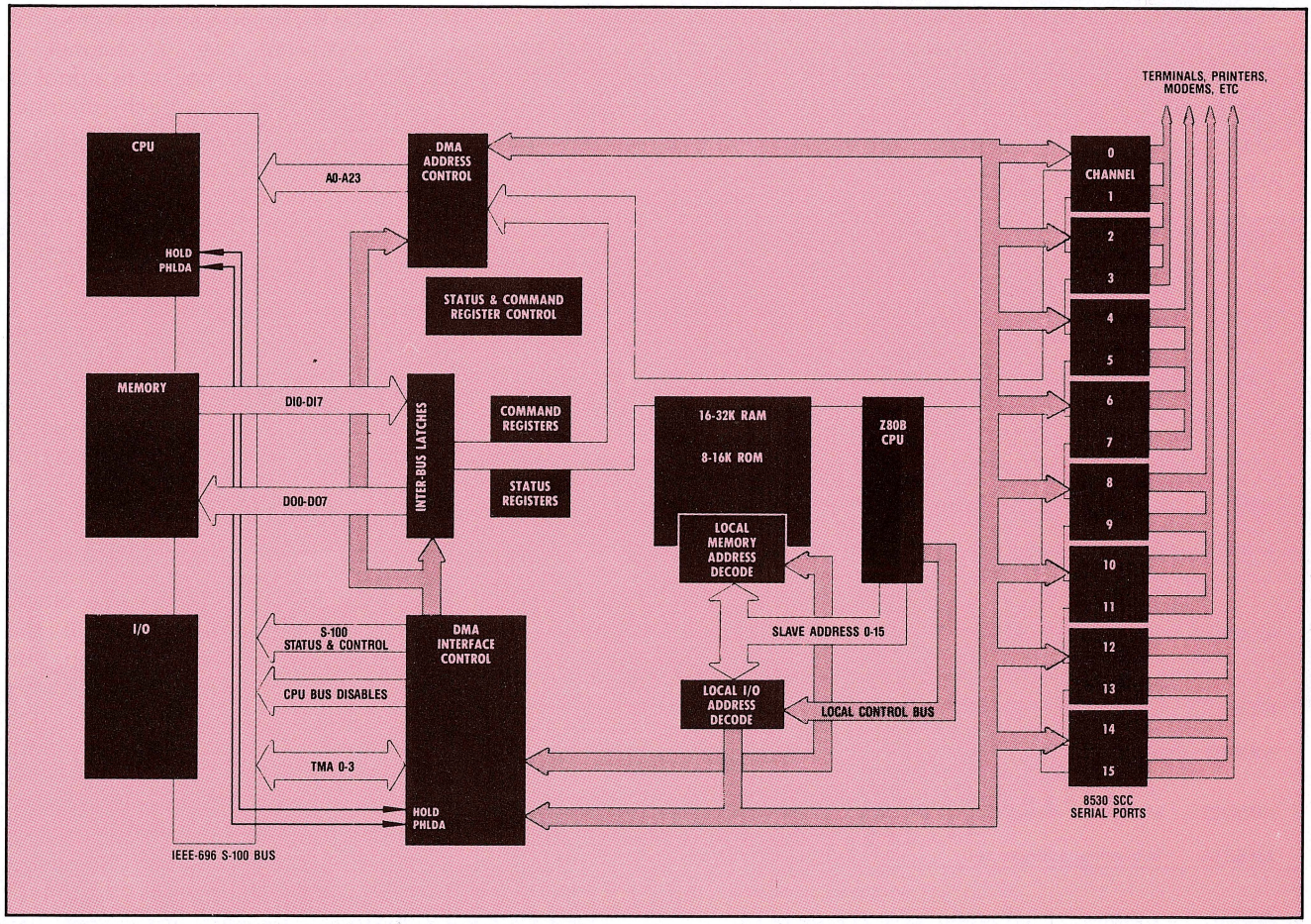


Figure 1. *Block diagram of the ADIT on-board components and their interface to the S-100 bus and peripherals.*

waste time waiting for the AOS's response. Instead, when the host later wants the status, it returns and checks the 'mailbox' registers.

Extended commands must pass more information than can be contained in the four interface registers. Therefore, they use an *Extended Command Control Block* (Table 2) set up in system memory to pass additional parameters and data. To initiate an extended command, the host issues an immediate command (AEXT) that includes a 24-bit address of the extended command block. For efficiency, the AOS interleaves commands (directed at any particular I/O channel) that might not be processed immediately with commands directed at other channels.

A special subset of the extended commands, M-channel commands, are used for memory-to-memory transfer of data and allow access to memory that is outside the host system address space. There is also a special subset of immediate commands, the V-channel commands, that are used to define and access a memory area as a virtual disk.

The ADIT board contains eight 8530 serial communications controllers with two channels on each controller. The extended command set includes the ability to issue primitive commands directly to the 8530. This gives the programmer complete freedom to use all capabilities of the 8530, such as establishing synchronous communication with mainframes, setting the number of stop bits per character, even or odd parity, number of bits per character transferred, sending a break, disabling of lower chain bit to prevent lower-priority devices on daisy chain from requesting interrupts, and the ability to write or read directly to any 8530 register.

A number of options, such as parity, data bits, baud rate, XON/XOFF support, are jumper selectable. Any of these jumpered defaults can be dynamically reconfigured by the host at any time. For example, the ADIT could be programmed to transmit at one baud rate and receive at another baud rate on any given channel.

Sample I/O drivers for MP/M-86, CP/M-2.2, and AMOS operating systems are included in the ADIT documentation. Additional support programs are provided for the AMOS operating system with each board purchased.

DYNAMIC ALLOCATION OF ON-BOARD MEMORY

The AOS, a multitasking, real-time operating system, resides in 8K of ROM. All 16 channels have access to 16K or 32K of RAM — divided into discrete 32-byte packets. Each packet contains 4 bytes of chain-pointer information and 28 bytes of data storage space. If a message is longer than 28 characters, the AOS links additional packets.

ADIT IMMEDIATE COMMANDS

ASTAT	Request channel status	*Gives current channel status, including 8530 status bits.*
ARSET	I/O channel reset	*Recovers host control of I/O channel when interrupt service has been requested but has not occurred. Can also be used to abort current I/O or clear all conditions to prepare for reconfiguring channel.*
AERSET	I/O channel error reset	*If host drivers implement parity error detection, lost data recovery, or break monitoring, this command can be used to clear these latched status flags.*
ARECN	I/O channel reconfiguration	*Used to modify a channel's parameters*
ASDI	Single data character input	*Obtains a single input data character.*
AISDA	Interrupt, data available	*Notifies host that single character input data is available.*
ASDO	Single data output	*Transfers a single output data byte.*
AISDOB	Interrupt, output space	*Notifies the host when buffer space is available for single character output.*
ASPCL	Special conditions	*This command is a collection of six subcommands that allow special control and cancellation processes.*
AMEMTST	Nondestruct memory test	*Reports if any errors are found after one nondestructive pass of ADIT RAM.*
AMULTI	Multichannel input status	*Sets flag bits corresponding to each channel to indicate if input data was received and is in ADIT input buffer.*
AMULTO	Multichannel output status	*Flag bits set to indicate ADIT output buffer has space to accept additional output character(s).*
AFRSET	Full board reset	*Full reset of ADIT board without requiring full host reset.*
AEXT	Set up extended command	*Sets up registers with address of the Extended Command Block (ECB). The extended command to be performed is located in the ECB.*
AEXTI	Set up ext com w/ interrupt	*Same as AEXT, with interrupt on completion.*

ADIT M-CHANNEL AND V-CHANNEL COMMANDS

AMSTAT	M-channel request status	*Returns current status of the M-channel (used for memory-to-memory transfers).*
AMMVE	M-channel extended move	*For utility use by host operating system. Gives software access to memory that may not normally be within the host operating system address space.*
AMMVEI	M-chan ext mv w/ interrupt	*Same as AMMVE, except interrupt is generated when move is completed.*
AMSETA	M-channel address set	*Stores the M-channel (memory channel) control block address permanently, so that it is used by future M-channel commands.*
AVSTAT	V-channel request status	*Returns status for the virtual disk (vdisk) channel.*
AVSETM	V-channel address set	*Issues 24-bit address of the start of memory to be used for virtual disk operations.*
AVSETP	V-channel set parameters	*Sets parameters for vdisk operations, including size of vdisk area, sector size, and number of sectors per track.*
AVSETC	V-channel current sector	*Issued before a fill command to give desired track and sector number.*
AVSETD	V-channel set DMA addr	*Sets address of buffer area used to send or receive data from the vdisk memory area.*
AVFILL	V-channel fill vdisk	*Aids in formatting vdisk by filling the vdisk memory locations with the specified fill byte value.*
AVRD	V-channel read	*Transfers data from the vdisk to the host DMA buffer area.*
AVRDI	V-channel read w/ interrupt	*Same as AVRD, but with interrupt generated when read is complete.*
AVWR	V-channel write	*Transfers data from host DMA buffer area to vdisk.*
AVWRI	V-channel write w/ interrupt	*Same as AVWR, but interrupt is generated when write is complete.*

ADIT EXTENDED COMMANDS

SYNRCON	Channel reconfiguration	*Reinitializes channel, outputting values to 8530 registers.*
SETECBA	Set ECB address	*Permanently stores the Extended Command Block (ECB) address for future ECB commands.*
OUTWBC	Output with byte count	*Moves a message of specified length from output buffer to channel via DMA.*
OUTWEOM	Output with EOM	*Moves a message terminated by a specified EOM (end of message) from output buffer to channel via DMA.*
INWBC	Input with byte count	*Stores input characters in the host input buffer, via DMA, until the total number stored equals a specified byte ct.*
INWEOM	Input with EOM	*Same as INWBC, except input is terminated by EOM of specified value, rather than byte count.*

Table 3. *ADIT commands. Execution of immediate commands requires a command name and parameters to be placed in the ADIT registers. Extended commands require the transfer of more information than can be held in the ADIT registers, so the registers will contain only the location of the Extended Command Block (ECB) resident in system memory. The ECB then contains the command to be executed and information for its execution. The M-channel subset of commands allows memory-to-memory tranfer of data, and V-channel commands are used to define and access a virtual disk set in memory.*

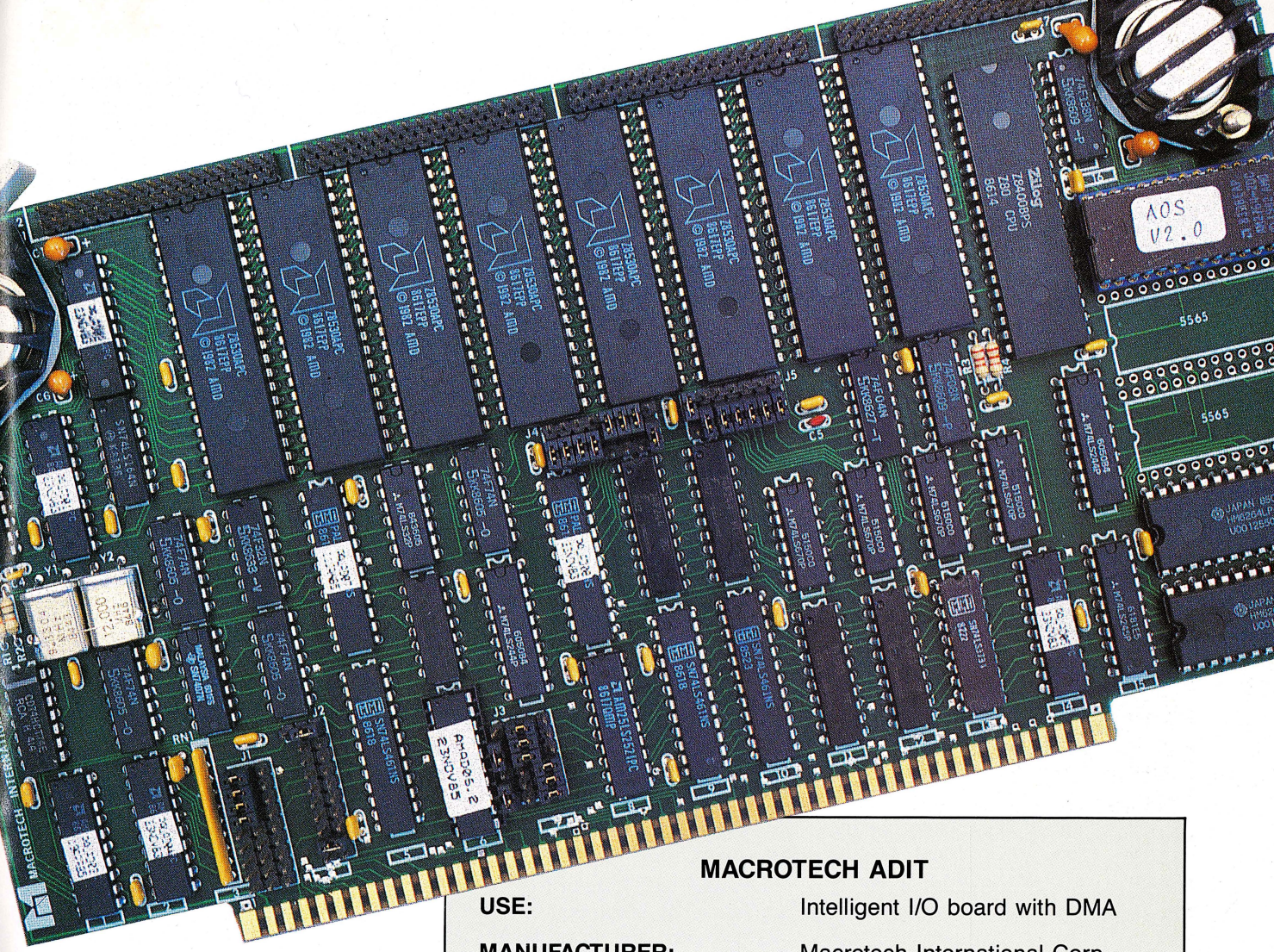
After the board transfers a group of 28 characters over a channel, it returns the empty packet to a pool of available packets. Each channel needing memory space obtains these packets from a pool. At any given instant, one of the 16 channels might not need memory or might need thousands of bytes of memory. The AOS dynamically allocates memory to the channels. This process is transparent to the host and to the user.

The ADIT board uses closed-loop feedback techniques and dynamic allocation of on-board memory to maintain control during heavy usage. When the board is handling communications on multiple channels, some channels will be doing output from the host while others will be active on input. If the capacity is nearing saturation, the AOS slows down output-character handling by allocating less of the available memory to the output channels while preventing data loss. However, the system must process input data as it occurs to prevent data loss or overrun conditions. The AOS maintains sufficient memory packets for input use so that input is not slowed and data is not lost. The AOS dynamically adjusts memory allocation as needed in the real-time environment to prevent data loss.

The AOS command set includes a nondestructive memory test for the ADIT RAM. This command will return a zero in two of the interface registers if no compare errors in its random memory test are found. The memory test may be run while all channels and all other commands are active.

PALS AND SIX-LAYER ARCHITECTURE CONCENTRATE LOGIC

The ADIT is a six-layer printed-circuit board. Programmed-array-logic (PAL) integrated circuits contribute to logic concentration. All decoding and creation of output terms and subterms can be implemented on PALs, eliminating the need for separate chips. The use of static RAM further reduces the



chip-count of the board by eliminating the need for additional refresh circuitry.

The Z80B-based ADIT contains eight 8530 controllers with two channels per controller. Each pair of 8530s connects to a single 34-pin ribbon connector that has four paddle cards, each providing the cable connection for a single channel. All channels support asynchronous I/O. In addition, four channels support synchronous communications, with the software for synchronous I/O to be supplied by the user. In a fully implemented ADIT board, there are four ribbons with four channels each, providing a total of 16 channels.

MACROTECH ADIT

USE:	Intelligent I/O board with DMA
MANUFACTURER:	Macrotech International Corp. 21018 Osborne Street, Bldng 5 Canoga Park, CA 91304 (818)-700-1501
COMMAND PORTS:	Five ports Address width: 8 bits Address select: 8-byte boundaries Data width: 8 bits
DIRECT MEMORY ACCESS:	Address width: 24 bits Data width: 8 bits
ON-BOARD PROCESSOR:	6-MHz Z80B, no wait states
ON-BOARD MEMORY:	ROM: 150ns, 8K or 16K RAM: 120ns, 16K or 32K, static
SERIAL CONTROLLERS:	Eight 6-MHz 8530 chips
SERIAL PORTS:	4, 8, 12, or 16. Two per controller. Synch or Async Transmission Flexible baud-rate generation On-board handshaking control
MANUAL:	104-page technical manual with schematics
RETAIL PRICE:	$985 to $1,795 (4 to 16 ports)

news and new products

Intelligent I/O Controller From Nu Com Corporation

The ACC8 S-100 intelligent I/O card is now available from Nu Com Corporation. This card features an on-board MC68008 processor and MC68681 DUART communications chips.

The board provides eight individually-controllable RS-232 serial ports for unidirectional or bidirectional communications. It comes with 8K of ROM (expandable to 16K), up to 32K of dual-ported RAM, and either a 8-MHz or 16-MHz clock. Device drivers are available for several hardware/software configurations.

The ACC8 is manufactured under the ERG trademark. Price is $730.00. For additional information, please send an Editorial Feature Reply Card to Nu Com Corporation, 19944 NE Ballinger Way, Seattle, WA 98155. ★

CCT Offers Software Selectable Serial/Parallel I/O Buffer Board

From Compliance Computer Technology is available the Printerfacer 1. This is an S-100 I/O board that features up to 1 Megabyte of on-board buffer memory and a Z80 controller.

The Printerfacer 1 accepts up to 1 Megabyte of data, releases the CPU from the I/O operation, and takes care of the handshaking with a printer or other peripheral device. Both a serial and a parallel port are provided. Selection between the ports is by means of a simple ASCII code.

CCT offers assistance in configuring the board for your S-100 system. The company is now selling the Printerfacer 1 at the introductory price of $249 with 256K of dynamic RAM installed. Contact or send EF Reply Card to Compliance Computer Technology, CCT Bldg, 6476 Airpark Drive, Prescott, AZ 86301. ★

New S-100 Board Products From CompuPro

CompuPro (Viasyn Corporation) has produced new versions of several of its S-100 products, including a 10-MHz 8085/88, a new version of the Interfacer 4, and a Disk 1B.

New products also include a special-features board with a SCSI interface, an intelligent I/O board, and a 32-bit 68020 CPU board with 32-bit-wide data path.

Because of CompuPro's current marketing de-emphasis on board-level products, the recent boards are primarily intended for the OEM market. For information, please contact CompuPro, 26538 Danti Court, Hayward, CA 94545-3999. ★

Inner Access Releases 32-Bit 68020 Piggy-Back Card

Inner Access now offers the Powerhouse, a piggy-back board with a 32-bit 68020 processor and 68881 math coprocessor.

The new board plugs into the socket of the 68000 chip on the company's 68000 S-100 CPU board.

Retail price for the Powerhouse board is $1595. The 68000 board lists for $695. For more information, contact or send an EF Reply Card to Inner Access Corporation, PO Box 888, Belmont, CA 94002. ★

Vision Computers Introduces 80286 S-100 Slave Processor Board

The GPS-286 is a new processor board that is compatible with CompuPro and Macrotech master CPU boards running Concurrent DOS. The 8-MHz, 80286-based GPS-286 allows up to 16 users or application programs per board; a total of 8 boards can be installed in a system.

Each board comes with 1 Megabyte of RAM and auto-install software drivers. Retail price is $1,500.

For more information, please contact or send an Editorial Feature Reply Card to Vision Computers, 2235 Melvin Road, Oakland, CA 94602. ★

1-Megabyte Version Of LS-100 Now Available From Digital Research Computers

Digital Research has released a new version of the Light-Speed-100 RAM-disk. This new disk-emulator board features 1 Megabyte of dynamic RAM and I/O port selection.

Software drivers are supplied to run the board in a CP/M-2.2 environment. Up to four boards can be cascaded for up to 4 Megabytes of RAM-disk.

The 1-Meg LS-100 is available in kit form for $259 or fully assembled and tested for $309. A blank printed circuit board with CP/M-2.2 drivers is $59. Please contact or send EF Reply Card to Digital Research Computers, PO Box 381450, Duncanville, TX 75138. ★

Zilog Z800 Samples Available To Developers

If you are a hardware developer trying to decide what next S-100 board to bring to market, you might consider a Z800 CPU board. The chip is now available to most OEMs, with full production expected this summer. The Z800 is a 16-bit superset of, and upwards compatible with the Z80 CPU. Among other features, it can address 16 Megabytes of memory.

As with other significant S-100 boards, S-100 Journal will provide front-cover exposure to the first full-functional true Z800 S-100 board that is loaned to us.

For information on obtaining a chip, contact Zilog, Inc., 1315 Dell Ave., Campbell, CA 95008, (408) 370-8283. ★

CP/M Software

FORMS-4 is a program designed to make filling out preprinted forms an easier task. It allows designation of fields with data types. It sells for $49.95. From Elliam Associates, 6101 Kentland Avenue, Woodland Hills, CA 91367. ★

REL/MAC is a utility program that will convert .REL files to assembly language source code files. One version is available to create Intel 8080 mnemonics and another to create Zilog Z80 mnemonics. Cost is $74.95 and $99.95 respectively. From Micro-Smith Computer Technology, PO Box 1473, Elkhart, IN 46515. ★

New Laser Printer From Ricoh Emulates HP LaserJet Plus

Ricoh Corporation recently unveiled the Laser 6000, a six-page-per-minute laser printer with 1 Megabyte of RAM. The Laser 6000 emulates the HP LaserJet Plus, Diablo 630, and Epson FX-80, among others.

Front-panel controls for common print and format functions are standard. The printer comes with both a Centronics parallel and RS-232 serial interface. Up to 32 fonts are supported, including eight resident fonts.

The Laser 6000 sells for $2395. Contact or send EF Reply Card to Ricoh Corporation, 5 Dedrick Place, West Caldwell, NJ 07006. ★

Medium-Cost Printer From Star Micronics Has Convenient Features

Star Micronics introduced a new dot-matrix printer, the ND-10, that provides 180-CPS and near-letter-quality modes. The ND-10 features a touch pad control panel that permits instant selection of commonly-used functions without the need for DIP-switch or software configuration.

A built-in tractor and single-sheet handling capabilities are standard. The ND-10 accepts both 10-inch fanfold paper and letter-size single sheets. Other features include downloadable character sets and Epson FX graphics emulation. Retail price is $499. Other related N-series printers are also available, including the ND-15 for 15-inch fanfold paper.

For additional information, write or send EF Reply Card to Star Micronics, 200 Park Avenue, Suite 3510, New York, NY 10166. ★

Macrotech Lowers Prices And Enhances Warranty

Macrotech International has lowered the prices of most of their S-100 boards.

The company also added extra features to its 1-year warranty program. Under the new terms, if a board fails within the first 90 days after purchase, Macrotech will ship a replacement board within 24 hours after receiving the defective board. If a board fails in the 91-to-365-day warranty period, the company will repair or replace a defective board within 5 days of receiving it. In either case, a RMA (Return Material Authorization) number must first be obtained from Macrotech before returning a defective board.

For more information about Macrotech's line of S-100 boards, which include all the former products from Octagon and SD Systems, send an EF Reply Card to Macrotech International Corp., 21018 Osborne Street, Bldg #5, Canoga Park, CA 91304. ★

Cochlin Computer Systems Offers DataFlex Seminars

Cochlin Computer Systems has released its schedule of DataFlex training seminars for the next three months. DataFlex is a database application development package that runs under Concurrent DOS, TurboDOS, and other operating systems.

Four types of seminars are offered. Level-1 seminars provide a working introduction to the language and are one-day long. Level-2 are two-day seminars that provide a more in-depth understanding of the language. DataFlex Macros is an advanced topics 2-day seminar for experienced DataFlex professionals. A 1-day File Relationships seminar is also offered.

Dates: Level-1, March 28 and May 16. Level-2, April 4-5 and June 6-7. Macros, April 25-26. File Relationships, April 11 and May 30. All dates are 1987. For additional information, send an EF Reply Card to Cochlin Computer Systems, 330 Townsend Street #107, San Francisco, CA 94107. ■

The News and New Products *section is compiled from press releases, information supplied by product manufacturers, and other news that might be of interest to the S-100 community. Information about S-100 products has first priority when considered for publication. High-quality photographs are welcome. All press releases should be dated and should include prices. Please send to New Products, S-100 Journal, 1275 N. University Ave., Unit 7, Provo, UT 84604.*

Z sets you free!

WHO WE ARE
Echelon is a unique company, oriented exclusively toward your CP/M-compatible computer. Echelon offers top quality software at extremely low prices; our customers are overwhelmed at the amount of software they receive when buying our products. For example, the Z-Com product comes with approximately 80 utility programs; and our TERM III communications package runs to a full megabyte of files. This is real value for your software dollar.

ZCPR3
Echelon is famous for our operating systems products. ZCPR3, our CP/M enhancement, was written by a software professional who wanted to add features normally found in minicomputer and mainframe operating systems to his home computer. He succeeded wonderfully, and ZCPR3 has become the environment of choice for "power" CP/M users.

Multiple Commands per Line
You can easily use multiple commands per line under ZCPR3. Simply separate the individual commands with semicolons. For example, "PIP B:=A:*.TXT;STAT B:*.*" will copy files and then show you the STAT results.

User-Programmed menu systems
ZCPR3 comes with three different menu systems that you can use to create custom menu-driven "front ends" for your computer. This is especially useful for setting up menus for your spouse or co-workers to use the computer, as they never have to see the A> prompt. All they have to do is press a single key to run any single or multiple CP/M programs, and when the task is done, control is automatically returned to the menu (ordinary CP/M menu programs cannot do this).

Extended Command Processing
When you type a command under CP/M, it will only look for the program in the current drive and user area. ZCPR3 gives you more flexibility by additionally searching other disks and user areas when resolving commands. You have full control of this function, called the PATH. This is probably the one element of ZCPR3 that is missed most if you return to "ordinary" CP/M.

Also, ZCPR3 supports the capability of grouping all your commonly used utility programs into a library file (*.LBR). This is great for systems with a small number of directory entries per disk, as the library file only uses one entry. It also has the advantage of reducing disk space requirements for a given set of programs, allowing you to put more programs on a disk. And the programs in the library file are invokable from the command line just like any other program not in the library.

Other Features
There's much more to ZCPR3, like named directories, online help system, etc., but it can't be described on one page. If you would like more information, consider the books shown below.

Z-SYSTEM
Perhaps the only shortcoming of ZCPR3 is that it is not a complete replacement for CP/M. This is what the Z-System does. The Z-System contains ZCPR3 and an additional module, ZRDOS, and is a complete replacement for CP/M. ZRDOS adds even more utility programs, and has the nice feature of no need to warm boot (^C) after changing a disk. Hard disk users can take advantage of ZRDOS "archive" status file handling to make incremental backup fast and easy. Because ZRDOS is written to take full advantage of the Z80, it executes faster than ordinary CP/M and can improve your system's performance by up to 10%.

INSTALLING ZCPR3/Z-SYSTEM
Echelon offers ZCPR3/Z-System in many different forms. For $44 you get the complete source code to ZCPR3 and the installation files. However, this takes some experience with assembly language programming to get running, as you must perform the installation yourself.

For users who are not qualified in assembly language programming, Echelon offers our "auto-install" products. Z-Com is our 100% complete Z-System which even a monkey can install, because it installs itself. Z-Com includes many interesting utility programs, like UNERASE, MENU, VFILER, and much more.

Echelon also offers "bootable" disks for some CP/M computers, which require absolutely no installation, and are capable of reconfiguration to change ZCPR3's memory requirements. At present, only Kaypro computers have this option available.

BOOKS
We sometimes joke around the office that we are really in the business of publishing, not selling software. We have books. Lots of books. We have to have lots of books, considering how powerful our software is and the large quantity of different packages we offer. Here are our best sellers:

ZCPR3: The Manual
This is the "bible" for the ZCPR3 user. An exhaustive technical reference, bound softcover, 350 pages. Contains descriptions of each ZCPR3 utility program, a detailed discussion about the innards of ZCPR3, and a full installation manual for those doing their own installation. You could order it from B. Dalton, but why? Get it from us.

The Z-System User's Guide
For those who are not technically inclined. This is an excellent tutorial-style manual filled with examples of how to use the power of ZCPR3/Z-System most effectively, written by two highly experienced Z users. (One user is a lawyer, the other a writer; this proves that anyone can use Z and benefit from it.)

ZCPR3: The Libraries
The extensive documentation for the libraries of ZCPR3, SYSLIB, Z3LIB, and VLIB. A must for any serious user of these programming tools. Loose-leaf notebook style; easy to work with as it will lay flat on your desk.

THERE'S MORE
We couldn't fit all Echelon has to offer on a single page (you see how small this type is). We haven't begun to talk about the many additional software packages and publications we offer. Send in the order form below and just check the "Requesting Literature" box for more information.

Item	Name	Price
1	ZCPR3 Core Installation Package	$44.00 (3 disks)*
2	ZCPR3 Utilities Package	$89.00 (9 disks)
3	Z3-Dot-Com (Auto-Install ZCPR3)	$99.00 (6 disks)*
4	Z3-Dot-Com "Bare Minimum"	$49.95 (1 disk)
5	Z-Com (Auto-Install Z-System)	$119.00 (7 disks)*
6	Z-Com "Bare Minimum"	$69.95 (2 disks)
12	PUBLIC ZRDOS Plus (by itself)	$59.50 (1 disk)
13	Kaypro Z-System Bootable Disk	$69.95 (3 disks)
20	ZAS/ZLINK Macro Assembler and Linker	$69.00 (1 disk)
21	ZDM Debugger for 8080/Z80/HD64180 CPU's	$50.00 (1 disk)
22	Translators for Assembler Source Code	$51.00 (1 disk)
23	REVAS3/4 Disassembler	$90.00 (1 disk)
24	**Special –** Items 20 through 23	$150.00 (4 disks)
25	DSD-80 Full Screen Debugger	$129.95 (1 disk)
27	The Libraries. SYSLIB, Z3LIB, and VLIB	$69.00 (8 disks)
28	Graphics and Windows Libraries	$49.00 (1 disk)
29	**Special –** Items 27, 28, and 82	$129.00 (9 disks)
40	Input/Output Recorder IOP (I/OR)	$39.95 (1 disk)
41	Background Printer IOP (BPrinter)	$39.95 (1 disk)
42	Programmable Key IOP (PKey)	$39.95 (1 disk)
43	**Special –** Items 40 through 42	$89.95 (3 disks)
60	DISCAT Disk cataloging system	$39.99 (1 disk)
61	TERM3 Communications System	$99.00 (6 disks)
64	Z-Msg Message Handling System	$99.00 (1 disk)
81	**ZCPR3: The Manual** bound, 350 pages	$19.95
82	**ZCPR3: The Libraries** 310 pages	$29.95
83	Z-NEWS Newsletter, 1 yr subsciption	$24.00
84	**ZCPR3 and IOPs** 50 pages	$9.95
85	**ZRDOS Programmers's Manual** 35 pages	$8.95
88	**Z-System User's Guide** 80 page tutorial	$14.95

*Includes **ZCPR3: The Manual**

Echelon, Inc.
885 N. San Antonio Road, Los Altos, CA 94022 USA
415/948-3820 (order line and tech support)

NAME _____
ADDRESS _____
_____
TELEPHONE _____ DISK FORMAT _____
☐ REQUESTING LITERATURE

ORDER FORM

Payment to be made by:
☐ Cash
☐ Check
☐ Money Order
☐ UPS COD
☐ Mastercard/Visa:
#_____
Exp. Date _____

California residents add 7% sales tax.
Add $4.00 shipping/handling in North America, actual cost elsewhere.

ITEM	PRICE
_____	_____
_____	_____
_____	_____
Subtotal	_____
Sales Tax	_____
Shipping/Handling	_____
Total	_____

S-100 software

$20 PER ISSUE BUYS THIS SPACE

List your software product here in the S-100 Software directory. All listings must be of software that runs on an S-100 system. MS/PC-DOS software must also be available for other operating systems in order to be accepted. Messages must have approx. 30 words or less. We do all typesetting free.

NIRVONICS, INC.

PO Box 5062, Plainfield, NJ 07061
(201) 561-2155

Software for CP/M and MS-DOS. Products include: UNIX-style utilities, database program, form completion tool and an arcade-style action game requiring no graphics hardware. *Orders for custom software taken.*

CO-SA, COMPUTER UND SYSTEME GmbH

Krischerstr. 70, D-4019 Monheim, West-Germany
Phone:02173-396170 Fax: 02173-52071 Telex: 8515836 amph d

We are looking for cooperation with Software manufacturers-authors to adapt and translate Software into German. We specialize in UNIX and C language, MS-DOS and C, and C-Basic.

ZeitlerSystems

9 Pajaro Way, Salinas, CA 93901

Redirect and Pipe input/output of DIR, FIND, SORT, MORE, ECHO, CLS, WC under Concurrent 4.1. $29.95 includes shipping, CA residents add tax. Specify format: 8″ SSSD, 5″ MS-DOS, inquire about others.

SCS

166 Wright Street, Staten Island, NY 10304
(718) 442-0665

DISMAL: table-driven disassembler you can configure for almost any processor and mnemonics. Preconfigured for Z80, all documentation and source included for total user customization. Specify disc format. $50 prepaid.

CREVELT COMPUTER SYSTEM, INC.

3111 S. Valley View E-103, Las Vegas, NV 89102
TurboDOS SYSTEMS (702) 362-6946

ACCESS CONTROL SYSTEM provides control & security. Remote users have call-back protection. $95.00 retail.
NETWORK COMMUNICATION PACKAGE provides software for dial-up master-to-master networking. $125.00 retail.

Affordable Engineering Software

CALL OR WRITE FOR FREE CATALOG
Circuit Analysis • Root Locus • Thermal Analysis • Plotter Drivers • Graphics • Signal Processing • Filter Design • Report Proofreader • Transfer Function Analysis.

BV Engineering VISA/MC
2200 Business Way Suite 207, Riverside, CA 92501 (714) 781-0252

NO SOURCE CODE?

REL/PAK converts M80 format REL files to Z80 and 8080 MAC files which can be edited and then assembled. All public and external symbols inserted. Free brochure available. $99.95 prepaid.

Microsmith Computer Technology
PO Box 1473, Elkhart, IN 46515
(800) 662-4070 or in Illinois (800) 942-7317

HOWARD PETERS

Box 2127, Rockford, IL 61131

JOB — the best CP/M tasking utility. Replaces old submit/xsub. **JOB** runs virtually ALL programs — xsub won't! Many other capabilities — keyboard entry, nesting, wildcard, list driven, IF, GOTO, more. Customize it — Turbo/ASM SOURCE CODE and utilities included! Information SASE. Program $15.00, 8 inch SSSD. Inquire about 5 inch formats.

RAM DISK SOFTWARE: for MORROW Z80 CPU

Latest CP/M 2.2 BIOS utilizes up to 1 Megabyte extended address RAM. $195.00

JACOBS COMPUTER SERVICES
P.O. Box 6092, Albany, CA 94706
(415) 724-2446

DataCOM Super Systems™

(813) 791-1454 modem 300/1200/2400 bps

Remote multi-user 140mb TurboDOS system with files for TurboDOS, CP/M, MP/M, and PC/MS-DOS users. Courier 2400 modems and 2 phone lines. $35.00 annual fee. Call & download application today.

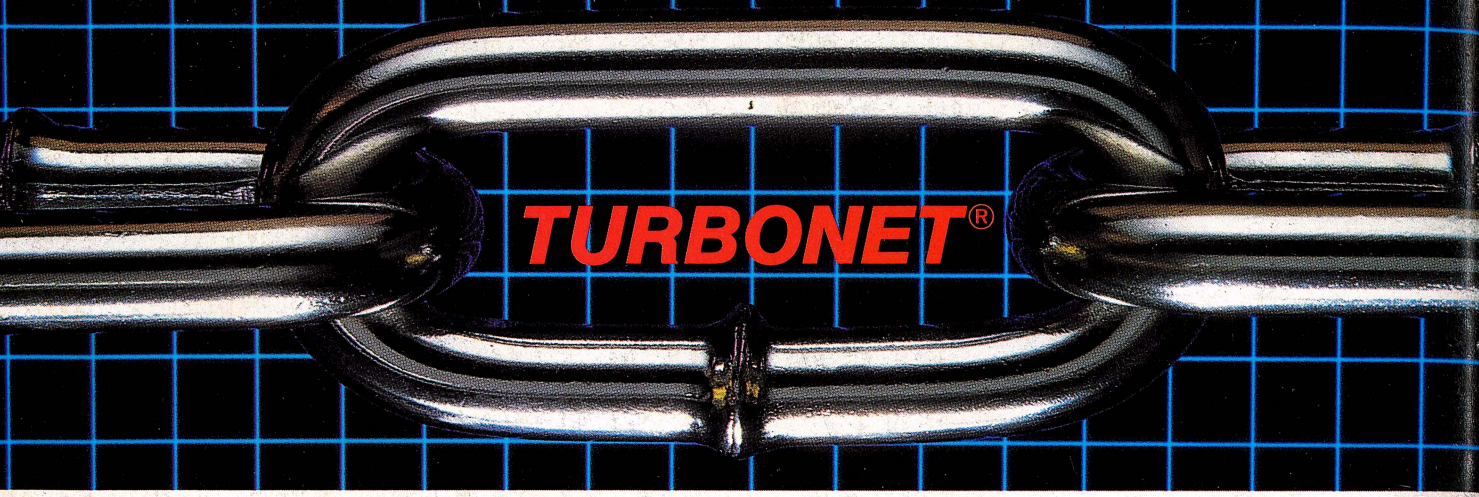
PC-PRO is PC-DOS for CompuPro/Viasyn

● PC DOS Ver 3.1 ● PC Graphics Support

for information write

Computer House, Inc.
PO Box 709, Woodacre, CA 94973 (415) 453-0865

Trademarks: Viasyn Corp. PC-PRO—Computer House



One Strong Link Can Strengthen Your Whole System

TurboNET® is Teletek's new local area network that permits reliable high speed data transfers among computers of differing bus architectures. TurboNET is designed to be used in networks consisting of multiple S-100 based and PC based systems. It will allow up to 4000 users, including up to 255 IBM PCs or compatibles, to share a single network and all attached peripherals. The network can be organized in any number of different ways mixing Teletek's 8 and 16-bit multiuser systems and PCs in any combination.

Teletek's Networking Family consists of:

TurboNET PC: IBM-PC Network Interface Board

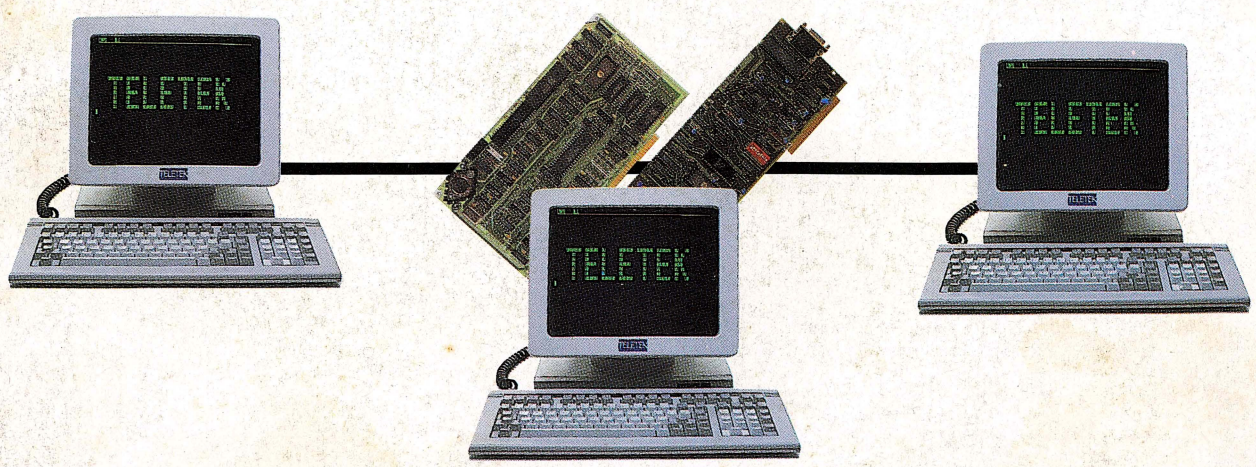
Teletek's TurboNET PC board offers IBM-PC Compatibility, CSMA industry standard protocols, 2 megabits/sec network speed, on-board CPU and communications management firmware, and media independence.

TurboNET S-100: S-100 Network Interface Board

Teletek's TurboNET S-100 board offers IEEE 696 Compatibility, CSMA industry standard protocols, 2 megabits/sec network speed, on-board CPU and communications management firmware, and media independence.

The benefits are obvious: The cost savings of shared peripherals, almost unlimited system expansion capability, and the use of existing PC workstations with the ability to run the myriad of application software written for MS-DOS and PC-DOS. This coupled with Teletek's 8 and 16-bit multiuser systems running application software written for CP/M and MP/M allows the system the ability to access almost any software library.

For more information on Teletek's TurboNET S-100 and TurboNET PC boards or on any of our full line of S-100 products, please call our Sales Department at 916-920-4600.



TELETEK

4600 Pell Drive, Sacramento, CA 95838
(916) 920-4600 Telex #4991834 Answer back — Teletek